GERMAN LINGUISTIC GUIDE

BY

LÉON GULIKERS
GILBERT RATTINK
RICHARD PIEPENBROCK

Unter all den Nebensachen der Welt ist die Rechtschreibung
jedoch eine der heikelsten. Was wir uns so mühsam aneignen,
wird uns ganz besonders teuer. Was wir automatisch zu beherrschen
lernen, wird sozusagen zu einem Teil der Person,
so daß uns jedes Ansinnen, daran etwas zu ändern,
fast wie eine Körperverletzung vorkommt.

— **DIETER E. ZIMMER** in Die Zeit 9th October 1992

Würde einer auf die Idee kommen, das Vokabularium,
das die meisten Eltern im Gespräch mit ihren Kindern verwenden,
einmal zu testen, würde er feststellen, daß das Vokabularium der >Bild<-Zeitung,
damit verglichen, fast das Wörterbuch der Brüder Grimm wäre.

— **HEINRICH BÖLL** (1917–1985) Ansichten eines Clowns (1963)

Hat jemand was verwirckt und bösen Lohn verdienet,
Den schicke ja nicht hin, daß er wird ausgesühnet
Ins Zucht- und Marterhaus—Galeeren sind zu schlecht—
Er schreib ein Wörter-Buch; so marterst du ihn recht.

— From a funeral oration delivered in 1675

CONTENTS

1 GERMAN ORTHOGRAPHY

Detailed and varied information is available on the orthographic forms of lemmas (both headwords and stems) and wordforms. You can choose from a range of transcriptions: they can be syllabified or unsyllabified, they can include or omit *diacritics* (as explained below), or, in some cases, they come with the order of the letters reversed, or with the letters sorted alphabetically. In addition, there are columns which tell you the number of letters or syllables a particular transcription contains.

1.1 SPELLING

Before defining the specific spelling columns available with both of the German lexicon types, it's worth considering a few important general features which apply to many of the important columns, namely *diacritics* and *reversed transcriptions*. After that come the individual spelling columns themselves.

1.1.1 DIACRITICS

As you work your way down the `ADD␣COLUMN␣`menus, you can see that on several occasions the last menu in the series allows you to select transcriptions which contain–or omit–*diacritics*. Diacritics are the accents written above certain characters as a guide to pronunciation. In German, they are called "Umlaut", which means vowel mutation. Not only does the absence or presence of an Umlaut lead to different pronunciation of a word, it also often means that a word will have a different meaning. This is a permanent feature of German orthography, and thus included in the database. Likewise, when foreign words are given in the database, the correct markers accompany them: *Papiermaché*. The é appears to be the only diacritic of foreign origin to be found in the German database. The current version of the German database contains no other special characters than those listed below.

These special accented characters are eight-bit characters designed for use on certain DIGITAL terminals (the VT220 and newer terminals). If you use such a terminal, or can get your own terminal to emulate it, then you look at the diacritics columns with no problems at all. If you have a completely different terminal, you can still use diacritics columns by selecting the `MODIFY␣COLUMNS␣`option `CONVERT␣`to change the DIGITAL eight-bit codes to the form your terminal needs to produce the same diacritic characters.

To do this, you need a table of the DIGITAL eight-bit codes that CELEX uses, such as the one given in part 6 of the manual (the *Appendices*). In it you can find out the hexadecimal codes of the letters you need to convert. You also need a table of the codes your terminal uses to produce the same diacritical markers. The example that follows converts all the DIGITAL eight-bit codes that are used in the German database to their MS-DOS equivalents (as defined in the 1985 OLIVETTI MS-DOS User Guide). The characters which occur with diacritic markers are as follows: Ü, ü, Ä, ä, Ö, ö, ß, é. When you reach the `MODIFY␣CONVERSION␣`window which can be opened by choosing the option `CONVERT␣`in the window `MODIFY␣COLUMNS`, first select a column which contains transcriptions with diacritics, then type in the following string:

```
([\x20-\x7F]+␣
|\xDC%\x9A|\xC4%\x8E␣|\xD6%\x99|\xFC%\x81␣
|\xE4%\x84|\xF6%\x94␣|\xDF%\xE1|\xE9%\x82)*␣
```

Once installed, this pattern will convert all the diacritic characters whenever you `SHOW␣`or `EXPORT␣`a column. If you're new to the pattern matcher and its capabilities then it may appear very mysterious, but in fact it's straightforward. Read the next couple of paragraphs for a full explanation.

The first line indicates that one or more normal ASCII codes (those with hexadecimal values between 20 and 7F) are allowed.

The remaining lines indicate the changes that must be made to any 8-bit characters that occur. The pattern matcher uses the `%␣`sign to indicate a conversion: the element to the left of the `%␣`is converted to the element on the right. (This use of the `%␣`sign is different from the 'wildcard' function it has at other times.) The pattern matcher also uses the symbols `\x␣`to mean that the two characters which follow

form a hexadecimal code – thus in the DIGITAL eight-bit code `\xDC␣`actually means Ü␣ In the MS-DOS coding set, the same Ü␣character is represented by the code `\x9A`. So to tell the pattern matcher to convert from a DIGITAL Ü␣to an MS-DOS Ü␣you must type `\xDC%\x9A`.

So far, this accounts for one diacritic character. To convert all the diacritic characters, you have to add extra parts to the pattern as appropriate, until you end up with a pattern like the one above. Each element is separated by the OR marker `|`. The whole pattern comes between brackets followed by an asterisk at the end `(...)*`, which means 'the word may be made up of zero or more of the elements between the brackets'.

1.1.2 REVERSE TRANSCRIPTIONS

Transcriptions without diacritics are often available in *reverse order*; each item is given back to front. Thus *fallen* is given as *nellaf*. The reason for this is that with a draft lexicon, looking up word endings can be done much more quickly when you use reverse transcriptions.

1.2 SPELLING COLUMNS

This section sets out the columns with spellings available for each lexicon type. First there is a subsection on the headword transcriptions available with the lemma lexicon, followed by a subsection on wordform transcriptions.

1.2.1 TRANSCRIPTIONS FOR LEMMAS

The German lemma is always represented by the headword (as described in the *Introduction* section 2.7). When you choose a column which contains orthographic transcriptions of headwords, it is as if you are choosing the bold-type headwords in a dictionary. All the other columns in the database contain information specific to individual headwords, so the main function of the orthographic transcriptions is to identify any other information you look up - looking at a list of lemma frequency figures isn't meaningful unless you can see the lemmas they refer to. However, you may not always need to see the orthographic form of the headword: if you're looking

for phonetic transcriptions with certain interesting syllable-final characteristics, say, you may not be interested in the orthographic headword - in which case you needn't keep it *on view*, and you might even want to miss it out of your lexicon altogether.

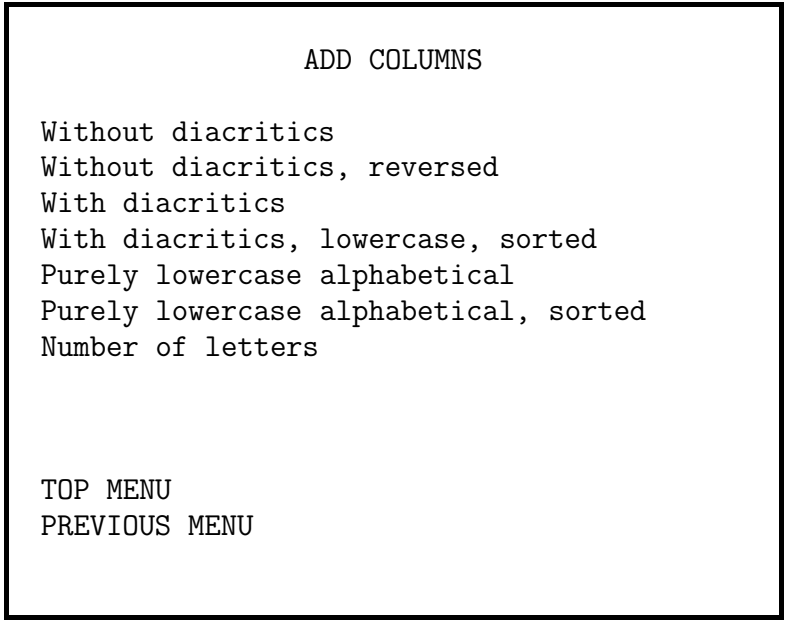
Described below are several different forms of orthographic transcriptions, and each form is assigned its own column. The first distinction you can make between them is whether or not syllable markers are included. Thereafter you can choose between back-to-front transcriptions which consist only of lower case characters, and even transcriptions with the letters of the headwords re-ordered alphabetically.

This FLEX window is the menu you see for a lemma lexicon when you choose the Orthography option of the first `ADD␣COLUMNS␣`menu, which is the first item in the option `MODIFY␣COLUMNS`:

```
                ADD COLUMNS

Headwords                                  >
Headwords,syllabified                      >
Stems                                      >
Stems, syllabified                         >



TOP MENU
PREVIOUS MENU
```

1.2.1.1 SPELLINGS FOR GERMAN HEADWORDS

There are seven columns offered in the `ADD␣COLUMNS␣`menus, and each contains spellings of headwords in a different form.



The first column contains information which is basic to the other six columns. It simply contains headwords composed of upper and lower case characters, with no diacritics or any other alterations. This means that the vowels ä, ö, ü, Ä, Ö, Ü and the 'sharp s' ß are replaced by the combinations ae, oe, ue, Ae, Oe, Ue and ss. The word *regelmäßig* is represented as *regelmaessig*. The FLEX name and description of this column are as follows:

Head `Headword␣`
(HeadLemma)

The second column contains the same transcriptions to be found in the first column, only the order of the letters is reversed. Thus the headword *Haus* is given as *suaH* and *Hoffnung* is given as *gnunffoH*. The word *ztesegsgnurehcis-revnetlletsegnA* can also be found in this column. The FLEX name and description of this column are as follows:

HeadRev `Headword,␣reversed␣`
(HeadRevLemma)

The third column gives spellings which include diacritics as well as the basic upper and lower case characters, hyphens and apostrophes of the basic transcriptions. So, while the first column gives the plain form *Gluehbirne*, this column includes the authentic "Umlaut": *Glühbirne*. The characteristics of diacritics are described in section 1.1.1 above. The

FLEX name and description of this column are as follows:

HeadDia `Headword,␣diacritics␣`
(HeadDiaLemma)

The fourth column contains lower case headwords with diacritics and their letters in alphabetical order. This column, which does not exist in the English and Dutch database, is important for German because two words may differ just because of these special characters, e.g. the lower case representation without diacritics for both the word *Maße* and the word *Masse* is the form *masse*. The sixth column in this window, which contains (purely lower case) headwords with their constituent letters in alphabetical order will therefore give one representation for these two words *aemss*. This fourth column, which also includes diacritics, will give *aemss* for the word *Masse*, whereas the word *Maße* will be represented as *aemß*. The FLEX name and description of this column are as follows:

HeadLowSortDia `Headword,␣lowercase,␣sorted,␣diacritics␣`
(HeadLowSortDiaLemma)

The next three columns use headwords with all upper case characters reduced to lower case characters and all diacritics removed without being replaced by e's as in the column *Headword*. This is particularly useful for automatic sorting programs: a column containing purely lower case alphabetical characters can be used to provide normal dictionary-like alphabetical order (i.e. not ASCII order, which differentiates between upper and lower case characters) for a lexicon, whatever the contents of its other columns.

The first of these three contains the ordinary headwords of the very first column with the upper case letters replaced by the corresponding lower case letters. The FLEX name and description of this column are as follows:

HeadLow `Headword,␣lowercase,␣alphabetical␣`
(HeadLowLemma)

The next column contains (purely lower case) headwords with their constituent letters in alphabetical order (*Abbaugerechtigkeit* becomes *aabbceeegghiikrttu*, for example). Using

this column, anagrams can be solved quickly, and searches for words containing certain numbers of letters can be carried out with ease: creating a query which looks for `aabb%␣`in this column can return a list of words (from another column) which contain two *a's* and at least two *b's*. The FLEX name and description of this column are as follows:

HeadLowSort `Headword,␣lowercase,␣alphabetical,␣sorted␣`
(HeadLowSortLemma)

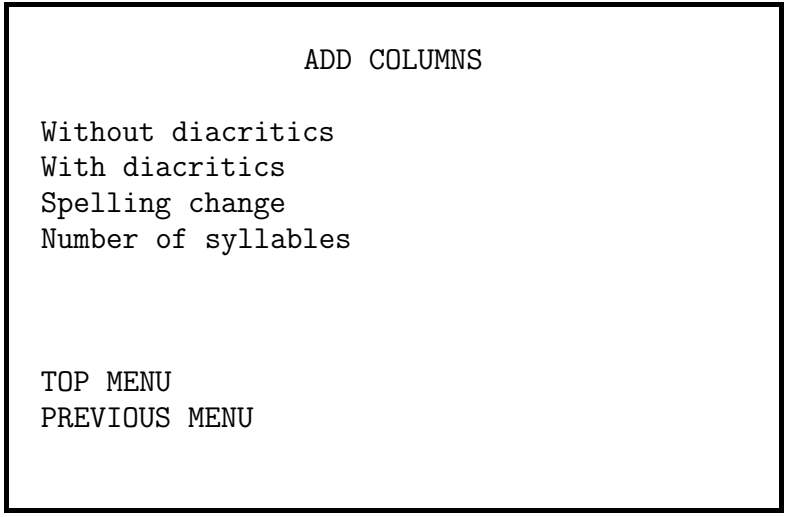
The seventh and last column contains counts of the number of letters in each headword. Here *letters* means any upper or lower case alphabetic characters with or without diacritics. This means that the number of letters in *abbröckeln* for example is 10. The FLEX name and description of this column are as follows:

HeadCnt `Headword,␣number␣of␣letters␣`
(HeadCntLemma)

1.2.1.2 SPELLINGS FOR SYLLABIFIED HEADWORDS

There are two columns which contain headwords with their orthographic syllable markers. In these columns, a hyphen marks the boundary between each pair of syllables within the headword. Thus the plain headword *Ablenkungsmanöver* is given as *Ab-len-kungs-ma-noe-ver* in the column `Without␣diacritics␣`and as *Ab-len-kungs-ma-nö-ver* in the second column `With␣diacritics`. The third column is a so-called Yes/No-column. It indicates whether hyphenation causes a change of one or more of the letters in the word or not. If for example the word *Abdeckung* is syllabified, this will lead to *Ab-dek-kung* or the word *Bettuch* which will be represented as *Bett-tuch*. In this third column this will be indicated as 'Y'.

There is a fourth column relating to syllabified headwords, and it tells you the number of orthographic syllables each headword has.



The first column contains the basic headwords plus syllable markers, each transcription consisting of upper and lower case characters, hyphens and apostrophes. The information about the place of hyphenation was taken from the *Duden Rechtschreibung der deutschen Sprache und der Fremdwörter (Mannheim 1986)* which is part 1 of the series of Duden lexicons. According to the Duden information it is not allowed for a syllable to contain one single character. To indicate a single vowel syllable boundary the = sign was introduced. It means that it is possible to place a syllable marker, although Duden's typografic conventions do not allow it. For example the word *Abendbrot* is presented here as *A=bendbrot*. Some people however like to use only *partially* syllabified headwords – that is, syllabified transcriptions which omit the syllable marker if the syllable consists of only one letter. For example, the partially syllabified transcription of *Abendbrot* would be *Abend-brot*. Such transcriptions are useful for automatic hyphenation programs, since typographic convention says that a word divided at the end of a line should consist of more than one character. To obtain transcriptions in this form, you can use the `CONVERT␣`option of the `MODIFY␣COLUMNS␣`menu. When you reach the `MODIFY␣CONVERSION␣`window, select a column containing normal syllabified headwords, and then type the following string:

`(=%|@)*␣`

This means: If a word contains a = sign, convert it into nothing and leave other characters as they are. Thus whenever you `SHOW␣`or `EXPORT␣`your lexicon, the syllabified transcriptions will always appear in partially syllabified form. For example the word *Abendbrot* will be shown as *Abend-brot*.

The FLEX name and description of this column are as follows:

HeadSyl (HeadSylLemma) `Headword, syllabified `

The second column contains the same headwords as the first, except that diacritics are included where appropriate. The FLEX name and description of this column are as follows:

HeadSylDia (HeadSylDiaLemma) `Headword, syllabified, diacritics `

As explained before, the third column is used to indicate whether the syllabification of a word causes certain characters to change. The FLEX name and description of this column are as follows:

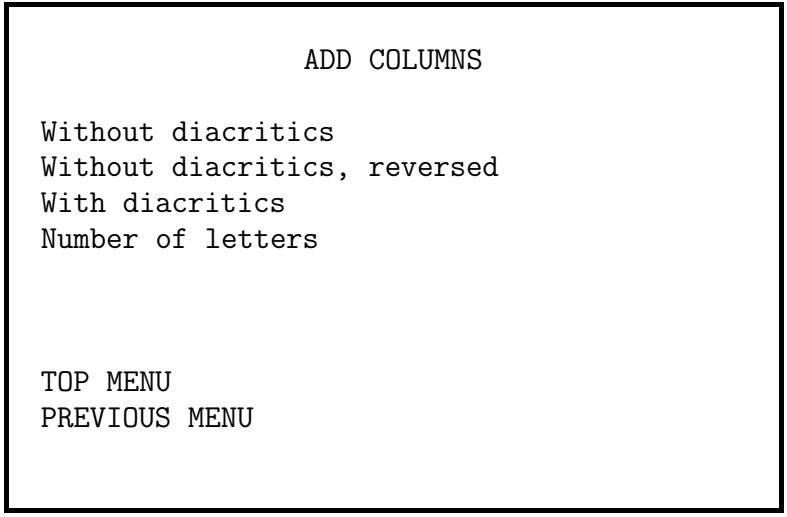
HeadSylChg (HeadSylChgLemma) `Spelling change, headword `

The fourth and last column for syllabified headwords tells you how many syllables each headword contains. Again the Duden rules were used to determine the syllable boundaries. The number of syllables in the word *Abendbrot*, for example, is 2, since according to Duden the word should be syllabified as *Abend-brot*. The FLEX name and description of this column are as follows:

HeadSylCnt (HeadSylCntLemma) `Number of orthographic syllables `

1.2.1.3 SPELLINGS FOR STEMS

A stem is that form of a lemma which most linguists prefer to use in their work, since it is generally the shortest occurring form in a family of inflections. A full description of the properties of stems can be found in part one of the manual, the *Introduction*, under the section called *Lexicon types*. There are four columns offered in the `ADD COLUMNS` menus, and each contains spellings of stems in a different form.



The first column contains information basic to the other three columns. It simply contains stems composed of upper and lower case characters, hyphens and apostrophes, with no diacritics or any other alterations. This means that the vowels ä, ö, ü, Ä, Ö, Ü and the 'sharp s' ß are replaced by the combinations ae, oe, ue, Ae, Oe, Ue and ss. The word *abdämpfen* is represented as *abdaempf*. Remember that the `Headword␣`representation of this verb is *abdaempfen*. The FLEX name and description of this column are as follows:

Stem `Stem␣`
(StemLemma)

The second column contains the same stems as the first, except that the characters are given in reverse order. (This enables you to look for word endings more quickly and with greater ease.) The FLEX name and description of this column are as follows:

StemRev `Stem,␣reversed␣`
(StemRevLemma)

The third column contains the plain stem (containing upper and lower case letters, hyphens, and apostrophes) complete with diacritic markers (as described in section 1.1.1 above). The FLEX name and description of this column are as follows:

StemDia `Stem,␣diacritics␣`
(StemDiaLemma)

The fourth and last plain stem column contains counts of the number of letters in each stem. Here *letters* means any upper or lower case alphabetic characters including "Umlaut", excluding hyphens and apostrophes. This means that the number of letters in *regelmäßig* for example is 10. The FLEX name and description of this column are as follows:

StemCnt `Stem,␣number␣of␣letters␣`
(StemCntLemma)

1.2.1.4 SPELLINGS FOR SYLLABIFIED STEMS

There are two columns which contain stems with their orthographic syllable markers. In these columns, a hyphen marks the boundary between each pair of syllables within the stem. Thus the plain stem *Ablenkungsmanöver* is given as *Ab-len-kungs-ma-noe-ver* in the column `Without␣diacritics␣` and as *Ab-len-kungs-ma-nö-ver* in the second column `With␣diacritics`. The third column is a Yes/No-column. It indicates if hyphenation causes a change of one or more of the letters in the word. If for example the word *Abdeckung* is syllabified, this will lead to *Ab-dek-kung*. There is a fourth column relating to syllabified stems, and it tells you the number of orthographic syllables each stem has.

```
                ADD COLUMNS

Without diacritics
With diacritics
Spelling change
Number of syllables



TOP MENU
PREVIOUS MENU
```

The first column simply contains stems composed of upper and lower case characters, hyphens and apostrophes, with no diacritics. As described in section 1.2.1.2, boundaries allowed by the Duden conventions are indicated by a hyphen, whereas an equal sign '=' delimits a single-vowel syllable not normally allowed in writing. Some people however like to use only

partially syllabified stems – that is, syllabified transcriptions which omit the syllable marker if the syllable consists of only one letter. For example, the partially syllabified transcription of *Abendbrot* would be *Abend-brot*. Such transcriptions are useful for automatic hyphenation programs, since typographic convention says that a word divided at the end of a line should consist of more than one character. To obtain transcriptions in this form, you can use the `CONVERT␣`option of the `MODIFY␣COLUMNS␣`menu. When you reach the `MODIFY␣CONVERSION␣`window, select a column containing normal syllabified headwords, and then type the following string:

`(=%|@)*␣`

This means: If a word contains a = sign, convert it into nothing and leave other characters as they are. Thus whenever you `SHOW␣`or `EXPORT␣`your lexicon, the syllabified transcriptions will always appear in partially syllabified form. For example the word *Abendbrot* will be shown as *Abend-brot*.

The FLEX name and description of this column are as follows:

StemSyl `Stem,␣syllabified␣`
(StemSylLemma)

The second column contains the plain stem (containing upper and lower case letters, hyphens, and apostrophes) complete with diacritic markers (as described in section 1.1.1 above). The FLEX name and description of this column are as follows:

StemSylDia `Stem,␣syllabified,␣diacritics␣`
(StemSylDiaLemma)

As explained before, the third column is used to indicate whether the syllabification of a word causes certain characters to change. The FLEX name and description of this column are as follows:

StemSylChg `Stem,␣Spelling␣change␣`
(StemSylChgLemma)

The fourth and last column for syllabified stems tells you how many syllables each stem contains, again according to the Duden rules. For the word *A=bend-gym-na-si-um*, for

example, the number of syllables is 5. The FLEX name and description of this column are as follows:

StemSylCnt (StemSylCntLemma) `Stem,␣number␣of␣orthographic␣syllables␣`

1.2.2 TRANSCRIPTIONS FOR WORDFORMS

Wordforms are the words which we use in everyday speech and writing, the inflected forms of the stems and headwords listed in dictionaries and databases. A full description of the properties of wordforms can be found in part one of the manual, the *Introduction*, under the section called 'Lexicon types'. Transcriptions are available either with or without syllable markers.

1.2.2.1 SPELLINGS FOR WORDFORMS

There are seven columns offered in the `ADD␣COLUMNS␣`menus, and each contains spellings of wordforms in a different form.

```
                ADD COLUMNS

Without diacritics
Without diacritics, reversed
With diacritics
With diacritics lowercase, sorted
Purely lowercase alphabetical
Purely lowercase alphabetical, sorted
Number of letters



TOP MENU
PREVIOUS MENU
```

The first column contains information which is basic to the other six columns. It simply contains wordforms composed of upper and lower case characters, hyphens and apostrophes, with no diacritics or any other alterations. This means that the vowels ä, ö, ü, Ä, Ö, Ü and the 'sharp s' ß are replaced by the combinations ae, oe, ue, Ae, Oe, Ue and ss. The word *regelmäßig* is represented as *regelmaessig*. The FLEX name and description of this column are as follows:

Word `Word␣`

The second column contains all the wordforms to be found in the first column, except that the order of the letters is `reversed`. The FLEX name and description of this column are as follows:

WordRev `Word,␣reversed␣`

The third column gives spellings which include diacritics as well as the basic upper and lower case characters, hyphens and apostrophes of the basic transcriptions. The characteristics of diacritics are described in section 1.1.1 above. The FLEX name and description of this column are as follows:

WordDia `Word,␣diacritics␣`

The fourth column contains lower case wordforms with diacritics and their letters in alphabetical order. This column, which does not exist in the English and Dutch database, is important for German because two words may differ just because of these special characters, e.g. the lower case representation without diacritics for both the word *Maße* and the word *Masse* is the form *masse*. The sixth column in this window, which contains (purely lower case) headwords with their constituent letters in alphabetical order will therefore give one representation for these two words *aemss*. This fourth column, which also includes diacritics, will give *aemss* for the word *Masse*, whereas the word *Maße* will be represented as *aemß*. The FLEX name and description of this column are as follows:

WordLowSortDia `Word,␣lowercase,␣sorted,␣diacritics␣`

The next three columns all give wordforms with upper case characters reduced to lower case characters and any non-alphabetic characters (hyphens, apostrophes) removed. Also, all diacritics have been removed without being replaced by e's as in the column *Word*. This is particularly useful for automatic sorting programs: a column containing purely lower case alphabetical characters can be used to provide normal dictionary-like (i.e. not ASCII order, which differentiates between upper and lower case characters) for a

lexicon, whatever the contents of its other columns. The first of these three contains the ordinary wordforms of the very first column with the upper case letters replaced by the corresponding lower case letters. The FLEX name and description of this column are as follows:

WordLow `Word,␣lowercase,␣alphabetical␣`

The next column contains (purely lower case) wordforms with their constituent letters in alphabetical order (*abberiefest* becomes *abbeeefirst*, for example). Using this column, anagrams can be solved quickly, and searches for words containing certain numbers of letters can be carried out with ease: creating a query which looks for `abb%␣`in this column can return a list of words (from another column) which contain one `a␣`and at least two `b␣`characters. The FLEX name and description of this column are as follows:

WordLowSort `Word,␣lowercase,␣alphabetical,␣sorted␣`

The seventh and last column contains counts of the number of letters in each wordform. Here *letters* means any upper or lower case alphabetic characters including special characters like the sharp 's' and diacritic characters. This means that the number of letters in *regelmäßig* for example is 10. The FLEX name and description of this column are as follows:

WordCnt `Word,␣number␣of␣letters␣`

1.2.2.2 SPELLINGS FOR SYLLABIFIED WORDFORMS

There are two columns which contain wordforms with their orthographic syllable markers. In these columns, a hyphen marks the boundary between each pair of syllables within the headword. Thus the plain wordform *Ablenkungsmanöver* is given as *Ab-len-kungs-ma-noe-ver* in the column `Without␣diacritics␣`and as *Ab-len-kungs-ma-nö-ver* in the second column `With␣diacritics`. The third column is a Yes/No-column. It indicates if hyphenation causes a change of one or more of the letters in the word. If for example the word *Abdeckung* is syllabified, this will lead to *Ab-dek-kung*.

There is a fourth column relating to syllabified wordforms, and it tells you the number of orthographic syllables each wordform has.

```
                    ADD COLUMNS

Without diacritics
With diacritics
Spelling change
Number of syllables



TOP MENU
PREVIOUS MENU
```

The first column contains wordforms plus syllable markers, each transcription consisting of upper and lower case characters, hyphens and apostrophes, with no diacritics. As described in section 1.2.1.2, boundaries allowed by the Duden conventions are indicated by a hyphen, whereas an equal sign = delimits a single vowel syllable. Some people like to use only *partially* syllabified wordforms – that is, syllabified transcriptions which omit the syllable marker if the syllable consists of only one letter. For example, the partially syllabified transcription of *Abendbrot* would be *Abend-brot*. Such transcriptions are useful for automatic hyphenation programs, since typographic convention says that a word divided at the end of a line should consist of more than one character. To obtain transcriptions in this form, you can use the `CONVERT␣`option of the `MODIFY␣COLUMNS␣`menu. When you reach the `MODIFY␣CONVERSION␣`window, select a column containing normal syllabified wordforms, and then type the following string:

`(=%|@)*␣`

This means: If a word contains an '=' sign, convert it into nothing and leave other characters as they are. Thus whenever you `SHOW␣`or `EXPORT␣`your lexicon, the syllabified transcriptions will always appear in partially syllabified form. For example the word *Abendbrot* will be shown as *Abend-brot*.

The FLEX name and description of this column are as follows:

WordSyl `Word,␣syllabified␣`

The second column contains the same wordforms as the first, except that diacritics are included where appropriate. The FLEX name and description of this column are as follows:

WordSylDia `Word,␣syllabified,␣with␣diacritics␣`

As explained before, the third column is used to indicate whether the syllabification of a word causes certain characters to change. The FLEX name and description of this column are as follows:

WordSylChg `Spelling␣change,␣Word␣`

The fourth and last column for syllabified wordforms tells you how many syllables each wordform contains. Again the Duden rules were used to determine the syllable boundaries. The number of syllables in the word *Abendbrot*, for example, is 2, since according to Duden the word should be syllabified as *Abend-brot*. The FLEX name and description of this column are as follows:

WordSylCnt `Number␣of␣orthographic␣syllables␣`

2 GERMAN PHONOLOGY

Phonetic and phonological transcriptions are available for lemmas, stems and wordforms, along with the appropriate CV patterns, stress patterns, and phoneme and phonetic syllable counts. In addition, when you are using a wordform lexicon, you can get phonetic information (and other information too) about the lemmas of any wordforms you look at in the morphology `ADD␣COLUMNS␣` menus. The Duden *Aussprachewörterbuch* (Mannheim, 1974) was used as the basis for the phonetic transcriptions. However some allophonic phenomena had to be ignored leading to transcriptions that may range between a purely phonetic and a purely phonemic level. This is why it would probably be better to use the term *phonemic transcription*. The next table contains those allophones which are used in the Duden *Aussprachewörterbuch* and the phonemes that are used in the CELEX database instead. It sometimes happened that Duden mentioned more than one possible way of pronunciation. In these cases we decided to choose the first transcription of a number of possible transcriptions.

DUDEN	CELEX
ɐ	ər
m̩, n̩, l̩	əm, ən, əl
ʀ	r
ç	x
i, i̯	iː
y, y̆	yː
o, o̯	oː
u, u̯	uː
e	eː
ø	øː
ɛ̃	ɛ̃ː
œ̃	œ̃ː
ɑ̃	ɑ̃ː
õ	õː

Phonetic transcriptions are available for the wordforms, headwords and stems.

2.0.1 COMPUTER PHONETIC CHARACTER SETS

Four different sets of phonetic character codes are available from CELEX. The first three sets are SAM-PA, CELEX and CPA, and they can be thought of as computerized versions of IPA. They use standard ASCII codes—those which can be typed in and read on almost any terminal—to represent certain of the IPA characters. As far as possible, these sets have been designed to resemble IPA; a lot of the characters you type or read look like their IPA counterparts. As with IPA, diphthong and affricates are represented by writing the two appropriate characters next to each other, and long vowels are indicated by length markers. In some cases, however, these conventions can lead to ambiguity: are the two vowels shown next to each other *really* a diphthong, or are they in fact two separate vowels? To overcome such problems, there are columns which contain transcriptions with syllable markers, and also columns available which have a delimiter placed after each consonant, affricate, vowel, long vowel or diphthong. So, these sets of computer codes for phonetic transcription can provide a readable approximation of IPA, with extra provision made to overcome the possibility of ambiguity.

The first of these three sets is the SAM-PA set. It was developed in connection with a European Community research program, and it has been presented in the *Journal of the International Phonetic Association* (1987) 17 : 22, pp. 94–114 as a widely-agreed computer-readable phonetic character set suitable for use with Danish, Dutch, English, French, German and Italian. For technical reasons, the version of SAM-PA implemented by CELEX has to include one change: the \␣character (ASCII code 92) representing the 'half-open front rounded' vowel sound has been implemented as /␣ (ASCII code 47). The second is a set originally designed for use within CELEX. The third is CPA, the *Computer Phonetic Alphabet*, or *Esprit 291*, which was developed in the Ruhr Universität Bochum, Germany.

The fourth set is the DISC set, so called because it is a computer phonetic alphabet made up of distinct single characters. It is fundamentally different from the other three in that it assigns one ASCII code to each distinct phonological segment in the sound systems of Dutch, English and German. Here *segment* means a consonant, an affricate, a short vowel,

a long vowel or a diphthong. There are two main advantages to this set. First, it provides one character for one segment – in contrast to the other three sets which use extra characters for long vowels, affricates and diphthongs. Second, there is no possibility of ambiguous transcriptions. A diphthong is always shown as a diphthong, and two separate vowels in proximity to each other (say on either side of a syllable boundary) can thus no longer be confused with a real diphthong; an affricate is always shown as such, and not as two consonants. For both these reasons, those interested in processing phonetic transcriptions—as opposed to reading transcriptions in a character set that resembles the familiar IPA—may well choose transcriptions in this character set. Its most basic codes correspond to SAM-PA; all the SAM-PA codes which represent short vowels and consonants are included in this set. The remaining long vowels, diphthongs and affricates have been assigned codes not already in use for other purposes. The resulting character set thus does not look as elegant and IPA-like as the other three sets. However, if you are mainly interested in the computer processing of transcriptions, such æsthetic considerations might not be so important.

Clearly, you have a wide choice of transcriptions available to you. The type you choose will depend on the nature of the task you have in mind. For IPA-like readability and non-ambiguous transcriptions, use the SAM-PA, CELEX or CPA sets. For computer processing tasks which need one-character-to-one-segment-correspondence, use the DISC set. In Appendix II there is a table which sets out DISC and how it relates to Dutch, English and German.

The table on the next page lists the basic set of segments for German. Each line gives an IPA character alongside a word which exemplifies the sound and the equivalent characters in the four computer-usable sets available with CELEX.

IPA	example	SAM-PA	CELEX	CPA	DISC
p	**P**akt	p	p	p	p
b	**B**ad	b	b	b	b
t	**T**ag	t	t	t	t
d	**d**ann	d	d	d	d
k	**k**alt	k	k	k	k
g	**G**ast	g	g	g	g
ŋ	Kla**ng**	N	N	N	N
m	**M**aß	m	m	m	m
n	**N**aht	n	n	n	n
l	**L**ast	l	l	l	l
R, r	**R**attte	r	r	r	r
f	**f**alsch	f	f	f	f
v	**W**elt	v	v	v	v
s	Gla**s**	s	s	s	s
z	**S**uppe	z	z	z	z
	Schiff	S	S	S	S
	Genie	Z	Z	Z	Z
j	**J**acke	j	j	j	j
x,ç	Ba**ch**,i**ch**	x	x	x	x
h	**H**and	h	h	h	h
w	**w**aterproof	w	w	w	w
pf	**Pf**erd	pf	pf	pf	+
ts	**Z**ahl	ts	ts	C/	=
	Ma**tsch**	tS	tS	T/	J
	Gin	dZ	dZ	J/	_
iː	L**ie**d	i:	i:	i:	i
ː	Adv**a**ntage	A:	A:	A:	#
aː	kl**a**r	a:	a:	a:	a
ɔː	**A**llroundman	O:	O:	O:	$
uː	H**u**t	u:	u:	u:	u
ɜː	Teamw**or**k	3:	3:	@:	3
yː	f**ü**r	y:	y:	y:	y
ɛː	K**ä**se	E:	E:	E:	)
eː	M**e**hl	e:	e:	e:	e
øː	M**ö**bel	|:	q:	q:	|
oː	B**oo**t	o:	o:	o:	o
eɪ	N**a**tive	eI	eI	e/	1
aɪ	Sh**y**lock	aI	aI	a/	2
ɔɪ	Playb**oy**	OI	OI	o/	4
aʊ	All**ou**ndsportler	aU	aU	A/	6
ai	w**ei**t	ai	ai	a/	W
au	H**au**t	au	au	A/	B
ɔy	fr**eu**t	Oy	Oy	o/	X

Table 1: Computer codes for German phonetic transcriptions

IPA	example	SAM-PA	CELEX	CPA	DISC
ɪ	M**i**tte	I	I	I	I
ʏ	Pf**ü**tze	Y	Y	Y	Y
ɛ	B**e**tt	E	E	E	E
œ	G**ö**tter	/	Q	Q	/
æ	R**a**gtime	{	&	^/	{
a	h**a**t	a	a	a	&
	K**a**lev**a**l**a**	A	A	A	A
ʌ	Pl**u**mpudding	V	V	^	V
ɔ	Gl**o**cke	O	O	O	O
ʊ	P**u**lt	U	U	U	U
ə	B**e**ginn	@	@	@	@
œ̃ː	Par**fum**	/~:	Q~:	Q~:	^
æ̃	**Im**promptu	{~	&~	^/~	c
ɑ̃ː	Dét**en**te	A~:	A~:	A~:	q
æ̃ː	Bass**in**	{~:	&~:	^/~:	0
ɔ̃ː	Affr**on**t	O~:	O~:	O~:	~

Table 2: Computer codes for German phonetic transcriptions

2.1 PHONETIC TRANSCRIPTIONS

Phonetic transcriptions are available for lemmas (headwords and stems) and also for wordforms. They are written using the four computer phonetic alphabets described in the previous section. In addition, there are columns containing CV patterns, and also some phonological representations for stems in the CELEX and SAMPA computer phonetic alphabets.

2.1.1 LEMMA TRANSCRIPTIONS

The first choices you must make in your search for phonetic transcriptions concern the form of the lemma you want to use (headword or stem) and whether you want your transcription to contain stress markers and/or syllable markers:

```
                ADD COLUMNS

Headwords, plain                          >
Headwords, syllabified                    >
Headwords, syllabified, with stress       >
Stems, plain                              >
Stems, syllabified                        >
Stems, syllabified, with stress           >

TOP MENU
PREVIOUS MENU
```

The columns available with each of these options are described in full in the six subsections which follow. If you want to see how all these different types of transcriptions look, then consult table 3: it gives a couple of examples from all the columns described below so that you can see at a glance the differences between them.

2.1.1.1 TRANSCRIPTIONS FOR HEADWORDS

This first set of columns offers *plain* transcriptions – that is, transcriptions which do not have any syllable markers or stress markers, written in each of the four coding systems already described:

```
                ADD COLUMNS

SAM-PA character set
CELEX character set
CPA character set
DISC character set
Number of phonemes


TOP MENU
PREVIOUS MENU
```

However three of these columns have one special feature: *each phonetic segment ends with a delimiter*. Here a *segment* means a vowel, a consonant, a long vowel, a diphthong, or an affricate. Using a delimiter avoids any possibility of ambiguity between the two parts of a diphthong or an

affricate – something which FLEX requires when it is working on TOOLBOX␣options such as NEIGHBOURS␣or COHORTS. These delimiter transcriptions are available in the SAM-PA, CELEX, and CPA character sets. Delimiters are not given with DISC transcriptions since the unique single-character nature of that set obviates the need to delimit each segment in this way.

The first plain headword transcription column uses the SAM-PA character set, and full stops (.) as segment delimiters. The FLEX name and description of this column are as follows:

PhonSAM `Phonetic␣headword,␣SAM-PA␣character␣set␣`
(PhonSAMLemma)

The second column uses the CELEX character set, and full stops (.) as segment delimiters. The FLEX name and description of this column are as follows:

PhonCLX `Phonetic␣headword,␣CELEX␣character␣set␣`
(PhonCLXLemma)

The third column uses the CPA character set, and full stops (.) as segment delimiters. (Normally CPA uses full stops as syllable markers, but here of course, no syllable markers are used.) The FLEX name and description of this column are as follows:

PhonCPA `Phonetic␣headword,␣CPA␣character␣set␣`
(PhonCPALemma)

The fourth column uses the DISC set. No delimiters, syllable markers or stress markers are included, since each character equals one segment. The FLEX name and description of this column are as follows:

PhonDISC `Phonetic␣headword,␣DISC␣character␣set␣`
(PhonDISCLemma)

The last column in this subsection gives you counts of the number of phonemes in each headword. Here *phoneme* means the same as *segment* – one phoneme equals a vowel, a consonant, a long vowel, a diphthong, or an affricate. Thus

for the word *Abdecker* the number of phonemes is given as 7, while for *Abdeckerei* the number is 8. The FLEX name and description of this column are as follows:

PhonCnt `Headword,␣number␣of␣phonemes␣`
(PhonCntLemma)

2.1.1.2 TRANSCRIPTIONS FOR SYLLABIFIED HEADWORDS

This set of transcriptions uses the same basic transcriptions as the first set, except that instead of segment markers, there are characters that mark each phonetic syllable. These are the columns which contain syllabified phonetic transcriptions of each headword:

```
                ADD COLUMNS

SAM-PA character set
CELEX character set
CELEX character set, with brackets
CPA character set
DISC character set
Number of syllables

TOP MENU
PREVIOUS MENU
```

In most cases transcriptions are syllabified by putting a hyphen (or, in the case of CPA, a full stop) at every syllable boundary within each word. A second method, available with the CELEX character set, is to enclose each syllable within square brackets. The advantage of the brackets notation is that so-called 'ambisyllabic consonants' can be clearly identified. Ambisyllabic consonants are those consonants which come between two syllables, and which belong to both of those syllables. However since the two consonants are pronounced as one consonant, these two are represented by one character between square brackets. For example, the `[s]␣` in the transcription *[ap][bla[s]@n]* of *abblassen* is part of the second syllable and the third syllable, whereas the `[z]␣`in the transcription *[ap][bla:][z@n]* of *abblasen* belongs to the third syllable only.

The first syllabified headword transcription column uses the SAM-PA character set, and syllable boundaries within words are shown by hyphens. The FLEX name and description of this column are as follows:

PhonSylSAM **(*PhonSylSAMLemma*)** `Syllabified␣phonetic␣headword,␣SAM-PA␣character␣set␣`

The next two columns both use the CELEX character set. The first marks every syllable boundary within each transcription with a hyphen. The FLEX name and description of this column are as follows:

PhonSylCLX **(*PhonSylCLXLemma*)** `Syllabified␣phonetic␣headword,␣CELEX␣character␣set␣`

The other CELEX syllabified phonetic headword column uses the brackets notation as described above, and its FLEX name and description are as follows:

PhonSylBCLX **(*PhonSylBCLXLemma*)** `Syllabified␣phonetic␣headword,␣CELEX␣character␣set␣(brackets)␣`

The next column gives syllabified headword transcriptions in the CPA character set. Every syllable boundary within each word is marked by a full stop. The FLEX name and description of this column are as follows:

PhonSylCPA **(*PhonSylCPALemma*)** `Syllabified␣phonetic␣headword,␣CPA␣character␣set␣`

The fifth column uses the DISC character set. Here every syllable boundary within each word is marked by a hyphen. The FLEX name and description of this column are as follows:

PhonSylDISC **(*PhonSylDISCLemma*)** `Syllabified␣phonetic␣headword,␣DISC␣character␣set␣`

The last column in this subsection gives counts of the phonetic syllables which occur in each transcription. For example, both *abblasen* and *abblassen* contain 3 syllables. The FLEX name and description of this column are as follows:

SylCnt **(*SylCntLemma*)** `Headword,␣number␣of␣phonetic␣syllables␣`

2.1.1.3 TRANSCRIPTIONS FOR STRESSED AND SYLLABIFIED HEADWORDS

This set of columns gives syllabified transcriptions that also mark the points of primary stress in each headword. Some of the transcriptions may cause some confusion because they seem to contain two stress marks for primary stress. The word *abertausend* for example has been transcribed as `'a:.b@r.'tA/.z@nt␣`in CPA (the '-sign is used to mark a stressed syllable). This feature, which can also be found in Duden, indicates that the word can be stressed in different ways depending on the way the word is used in the sentence. This is also known as *stress shift*.

These are the columns you can choose from:

```
                ADD COLUMNS

SAM-PA character set
CELEX character set
CPA character set
DISC character set
Stress pattern


TOP MENU
PREVIOUS MENU
```

The first column uses the SAM-PA character set, and as well as using hyphens to mark syllable boundaries, these transcriptions show points of primary stress by means of the 'double quote' character ("). This character is placed immediately before a stressed syllable. The FLEX name and description of this column are as follows:

PhonStrsSAM `Syllabified␣phonetic␣headword,␣with␣stress␣`
(PhonStrsSAMLemma) `marker,␣SAM-PA␣character␣set␣`

The second column uses the CELEX character set, and as well as using hyphens to mark syllable boundaries, these transcriptions show the points of primary stress with an inverted comma (') immediately before the stressed syllable. The FLEX name and description of this column are as follows:

PhonStrsCLX `Syllabified␣phonetic␣headword,␣with␣stress␣`
(PhonStrsCLXLemma) `marker,␣CELEX␣character␣set␣`

The third column uses the CPA character set, including full stops to mark syllable boundaries, and these transcriptions show points of primary stress with an inverted comma (') immediately before the stressed syllable. The FLEX name and description of this column are as follows:

PhonStrsCPA `Syllabified␣phonetic␣headword,␣with␣stressmarker,␣`
(PhonStrsCPALemma) `CPA␣character␣set␣`

The fourth column uses the DISC character set, and along with hyphens to mark syllable boundaries, these transcriptions show points of primary stress with an inverted comma (') immediately before the stressed syllable. The FLEX name and description of this column are as follows:

PhonStrsDISC `Syllabified␣phonetic␣headword,␣with␣stress␣`
(PhonStrsDISCLemma) `marker,␣DISC␣character␣set␣`

The last column in this subsection contains a simple stress pattern for each headword. A *stress pattern* is a string which shows how each phonetic syllable is stressed in speech. Each syllable is represented by one numeric character: either `0␣` or `1`. `1␣`indicates that the syllable receives primary stress, and `0␣`that it does not receive primary stress. Thus the four-syllable word *Biologe* has the stress pattern `0010␣`and *Biologie* has the pattern `0001`. Note that patterns with more than one `1␣`can occur. The FLEX name and description of this column are as follows:

StrsPat `Headword,␣stress␣pattern␣`
(StrsPatLemma)

2.1.1.4 SOME EXAMPLE TRANSCRIPTIONS

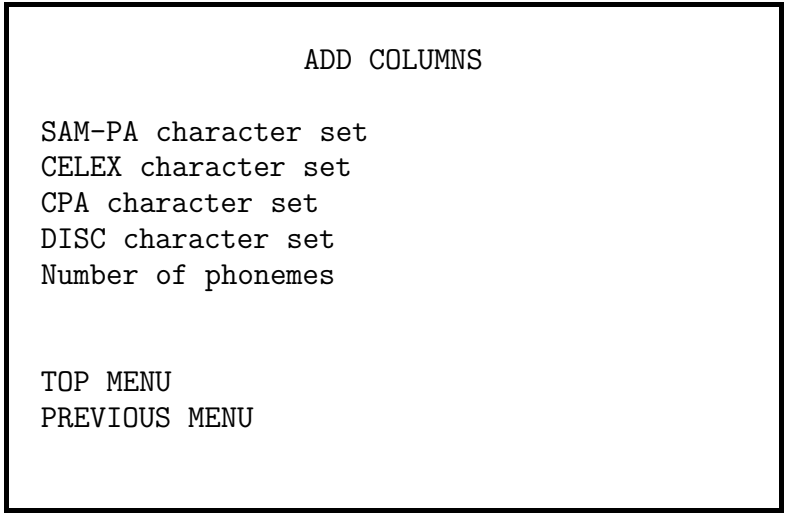
Column	Examples	
	abblasen	*abblassen*
PhonSAM	a.p.b.l.a:.z.@.n.	a.p.b.l.a.s.@.n.
PhonCLX	a.p.b.l.a:.z.@.n.	a.p.b.l.a.s.@.n.
PhonCPA	a.p.b.l.a:.z.@.n.	a.p.b.l.a.s.@.n.
PhonDISC	&pblaz@n	&pbl&s@n
PhonSylSAM	ap-bla:-z@n	ap-bla-s@n
PhonSylCLX	ap-bla:-z@n	ap-bla-s@n
PhonSylBCLX	[ap][bla:][z@n]	[ap][bla[s]@n]
PhonSylCPA	ap.bla:.z@n	ap.bla.s@n
PhonSylDISC	&p-bla-z@n	&p-bl&-s@n
PhonStrsSAM	"ap-bla:-z@n	"ap-bla-s@n
PhonStrsCLX	'ap-bla:-z@n	'ap-bla-s@n
PhonStrsCPA	'ap.bla:.z@n	'ap.bla.s@n
PhonStrsDISC	'&p-bla-z@n	'&p-bl&-s@n

Table 3: Example phonetic transcriptions

The table above lets you see the difference stress or syllable markers make to the appearance of your transcriptions. Use it in conjunction with the column descriptions to decide what sort of transcription you want to use. Although this table uses the names of the headword columns described above, the phonemic representations for stems are the same, except that the transcriptions for stems lack the infinitive ending.

2.1.1.5 TRANSCRIPTIONS FOR STEMS

This first set of columns offers *plain* transcriptions – that is, transcriptions which do not have any syllable markers or stress markers, written in each of the four coding systems already described:



However three of these columns have one special feature: *each phonetic segment ends with a delimiter*. Here a *segment* means a vowel, a consonant, a long vowel, a diphthong, or an affricate. Using a delimiter avoids any possibility of ambiguity between the two parts of a diphthong or an affricate – something which FLEX requires when it is working on TOOLBOX␣options such as NEIGHBOURS␣or COHORTS. These delimiter transcriptions are available in the SAM-PA, CELEX, and CPA characters sets. Delimiters are not given with DISC transcriptions since the unique single-character nature of that set obviates the need to delimit each segment in this way.

The first plain stem transcription column uses the SAM-PA character set, and full stops (.) as segment delimiters. The FLEX name and description of this column are as follows:

PhonStSAM `Phonetic␣stem,␣SAM-PA␣character␣set␣`
(PhonStSAMLemma)

The second column uses the CELEX character set, and full stops (.) as segment delimiters. The FLEX name and description of this column are as follows:

PhonStCLX `Phonetic␣stem,␣CELEX␣character␣set␣`
(PhonStCLXLemma)

The third column uses the CPA character set, and full stops (.) as delimiters. (Normally CPA uses full stops as syllable markers, but here of course, no syllable markers are used.) The FLEX name and description of this column are as follows:

PhonStCPA `Phonetic␣stem,␣CPA␣character␣set␣`
(PhonStCPALemma)

The fourth column uses the DISC set. No delimiters, syllable markers or stress markers are included, since each character equals one segment. The FLEX name and description of this column are as follows:

PhonStDISC `Phonetic␣stem,␣DISC␣character␣set␣`
(PhonStDISCLemma)

The last column in this subsection gives you counts of the number of phonemes in each stem. Here *phoneme* means the same as *segment* – one phoneme equals a vowel, a consonant, a long vowel, a diphthong, or an affricate. Thus for the word *Abdecker* the number of phonemes is given as 7, while for *Abdeckerei* the number is 8. The FLEX name and description of this column are as follows:

PhonStCnt `Stem,␣number␣of␣phonemes␣`
(PhonStCntLemma)

2.1.1.6 TRANSCRIPTIONS FOR SYLLABIFIED STEMS

This set of transcriptions uses the same basic transcriptions as the first set, except that instead of segment markers, there are characters that mark each phonetic syllable. These are the columns which contain syllabified phonetic transcriptions of each stem:

```
                ADD COLUMNS

SAM-PA character set
CELEX character set
CELEX character set, with brackets
CPA character set
DISC character set
Number of syllables

TOP MENU
PREVIOUS MENU
```

In most cases transcriptions are syllabified by putting a hyphen (or, in the case of CPA, a full stop) at every syllable

boundary within each word. A second method, available with the CELEX character set, is to enclose each syllable within square brackets. The advantage of the brackets notation is that so-called 'ambisyllabic consonants' can be clearly identified. Ambisyllabic consonants are those consonants which come between two syllables, and which belong to both of those syllables. For example, the `[b]␣`in the transcription *[a[b]re:][vi:][a[ts]i:][o:n]* of *Abbreviation* is part of the first syllable and the second syllable, whereas the `[b]␣`in the transcription *[ap][brEn]* of *abbrenn* belongs to the second syllable only.

The first syllabified stem transcription column uses the SAM-PA character set, and syllable boundaries within words are shown by hyphens. The FLEX name and description of this column are as follows:

PhonSylStSAM `Syllabified␣phonetic␣stem,␣SAM-PA␣character␣set␣`
(PhonSylStSAMLemma)

The next two columns both use the CELEX character set. The first marks every syllable boundary within each transcription with a hyphen. The FLEX name and description of this column are as follows:

PhonSylStCLX `Syllabified␣phonetic␣stem,␣CELEX␣character␣set␣`
(PhonSylStCLXLemma)

The other CELEX syllabified phonetic stem column uses the brackets notation as described above, and its FLEX name and description are as follows:

PhonSylStBCLX `Syllabified␣phonetic␣stem,␣CELEX␣character␣set␣`
(PhonSylStBCLXLemma) `(brackets)␣`

The next column gives syllabified stem transcriptions in the CPA character set. Every syllable boundary within each word is marked by a full stop. The FLEX name and description of this column are as follows:

PhonSylStCPA `Syllabified␣phonetic␣stem,␣CPA␣character␣set␣`
(PhonSylStCPALemma)

The fifth column uses the DISC character set, and here every syllable boundary within each word is marked by a hyphen. The FLEX name and description of this column are as follows:

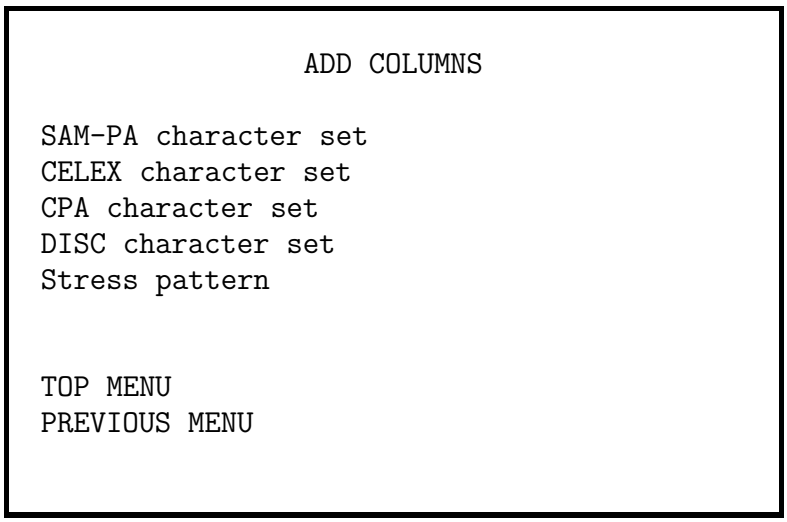
PhonSylStDISC `Syllabified␣phonetic␣stem,␣DISC␣character␣set␣`
(PhonSylStDISCLemma)

The last column in this subsection gives counts of the phonetic syllables which occur in each transcription. For example, both *abbitt* and *abbind* contain 2 syllables. The FLEX name and description of this column are as follows:

StSylCnt `Stem,␣number␣of␣phonetic␣syllables␣`
(StSylCntLemma)

2.1.1.7 TRANSCRIPTIONS FOR STRESSED AND SYLLABIFIED STEMS

This set of columns gives syllabified transcriptions that also mark the points of primary stress in each stem. These are the columns you can choose from:



The first column uses the SAM-PA character set, and as well as using hyphens to mark syllable boundaries, these transcriptions show points of primary stress by means of the ‘ double quote’ character ("). This character is placed immediately before a stressed syllable. The FLEX name and description of this column are as follows:

PhonStrsStSAM `Syllabified␣phonetic␣stem,␣with␣stress␣marker,␣`
(PhonStrsStSAMLemma) `SAM-PA␣character␣set␣`

The second column uses the CELEX character set, and as well as using hyphens to mark syllable boundaries, these transcriptions show the points of primary stress with an inverted comma (') immediately before the stressed syllable. The FLEX name and description of this column are as follows:

PhonStrsStCLX `Syllabified␣phonetic␣stem,␣with␣stress␣marker,␣`
(PhonStrsStCLXLemma) `CELEX␣character␣set␣`

The third column uses the CPA character set, including full stops to mark syllable boundaries, and these transcriptions show points of primary stress with an inverted comma (') immediately before the stressed syllable. The FLEX name and description of this column are as follows:

PhonStrsStCPA `Syllabified␣phonetic␣stem,␣with␣stress␣marker,␣`
(PhonStrsStCPALemma) `CPA␣character␣set␣`

The fourth column uses the DISC character set, and along with hyphens to mark syllable boundaries, these transcriptions show points of primary stress with an inverted comma (') immediately before the stressed syllable. The FLEX name and description of this column are as follows:

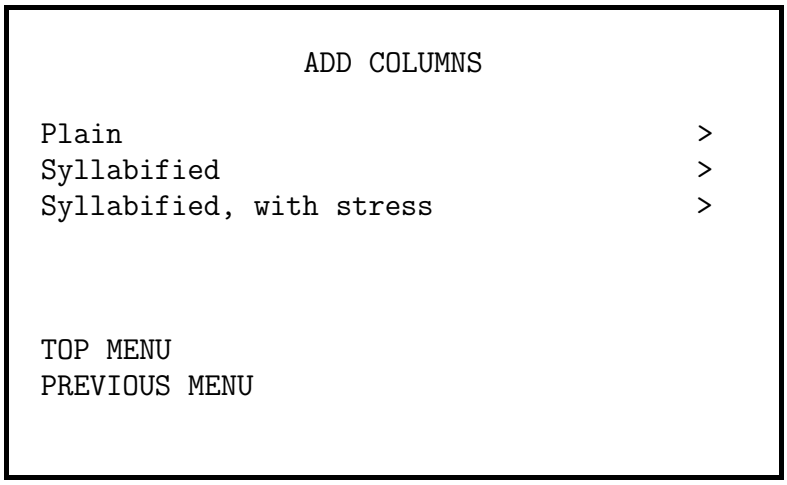
PhonStrsStDISC `Syllabified␣phonetic␣stem,␣with␣stress␣marker,␣`
(PhonStrsStDISCLemma) `DISC␣character␣set␣`

The last column in this subsection contains a simple stress pattern for each stem. A *stress pattern* is a string which shows how each phonetic syllable is stressed in speech. Each syllable is represented by one numeric character: either `0␣` or `1`. `1␣`indicates that the syllable receives primary stress, and `0␣`that it does not receive primary stress. Thus the four-syllable word *Biologe* has the stress pattern `0010␣`and *Biologie* has the pattern `0001`. Note that patterns with more than one 1 can occur. The FLEX name and description of this column are as follows:

StStrsPat `Stem,␣stress␣pattern␣`
(StStrsPatLemma)

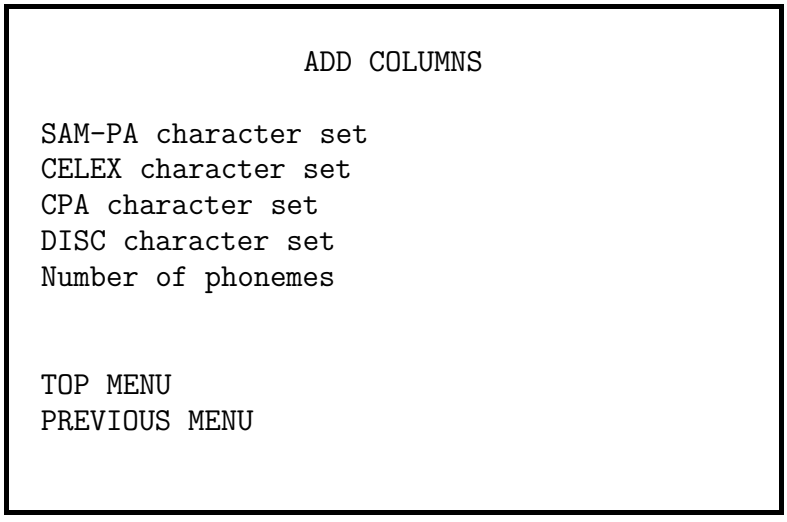
2.1.2 WORDFORM TRANSCRIPTIONS

A full range of phonetic transcriptions is available for wordforms. In addition, there are columns with phoneme and syllable counts and stress patterns for each wordform at appropriate points. You can choose them in your preferred computer phonetic character set, as described in section 2.0.1, but one small point to remember is that wordforms like *ahme nach* which include a space in their spelling also include a space in their phonetic transcription, thus for instance `a:.m.@.␣ n.a:.x.`. The first choice you have to make is whether you want plain transcriptions, syllabified transcriptions, or stressed and syllabified transcriptions:



2.1.2.1 TRANSCRIPTIONS FOR WORDFORMS

This first set of columns offers *plain* transcriptions – that is, transcriptions which do not have any syllable markers or stress markers, written in each of the four coding systems already described:



However three of these columns have one special feature: *each phonetic segment ends with a delimiter*. Here a *segment* means a vowel, a consonant, a long vowel, a diphthong, or an affricate. Using a delimiter avoids any possibility of ambiguity between the two parts of a diphthong or an affricate – something which FLEX requires when it is working on `TOOLBOX␣`options such as `NEIGHBOURS␣`or `COHORTS`. These delimiter transcriptions are available in the SAM-PA, CELEX, and CPA characters sets. Delimiters are not given with DISC transcriptions since the unique single-character nature of that set obviates the need to delimit each segment in this way.

The first plain wordform transcription column uses the SAM-PA character set, and full stops (.) as segment delimiters. The FLEX name and description of this column are as follows:

PhonSAM `Phonetic␣wordform,␣SAM-PA␣character␣set␣`

The second column uses the CELEX character set, and full stops (.) as segment delimiters. The FLEX name and description of this column are as follows:

PhonCLX `Phonetic␣wordform,␣CELEX␣character␣set␣`

The third column uses the CPA character set, and full stops (.) as delimiters. (Normally CPA uses full stops as syllable markers, but here of course, no syllable markers are used.) The FLEX name and description of this column are as follows:

PhonCPA `Phonetic␣wordform,␣CPA␣character␣set␣`

The fourth column uses the DISC set. No delimiters, syllable markers or stress markers are included, since each character equals one segment. The FLEX name and description of this column are as follows:

PhonDISC `Phonetic␣wordform,␣DISC␣character␣set␣`

The last column in this subsection gives you counts of the number of phonemes in each wordform. Here *phoneme* means the same as *segment* – one phoneme equals a vowel, a consonant, a long vowel, a diphthong, or an affricate. Thus for the word *ahme nach* the number of phonemes is given as 6, while for *ahmten nach* the number is 8. The FLEX name and description of this column are as follows:

PhonCnt `Wordform,␣number␣of␣phonemes␣`

2.1.2.2 TRANSCRIPTIONS FOR SYLLABIFIED WORDFORMS

This set of transcriptions uses the same basic transcriptions as the first set, except that instead of segment markers, there are characters that mark each phonetic syllable. These are the columns which contain syllabified phonetic transcriptions of each wordform:

```
                ADD COLUMNS

SAM-PA character set
CELEX character set
CELEX character set, with brackets
CPA character set
DISC character set
Number of syllables

TOP MENU
PREVIOUS MENU
```

In most cases transcriptions are syllabified by putting a hyphen (or, in the case of CPA, a full stop) at every syllable boundary within each word. A second method, available with the CELEX character set, is to enclose each syllable within square brackets. The advantage of the brackets notation is that so-called 'ambisyllabic consonants' can be clearly identified. Ambisyllabic consonants are those consonants which come between two syllables, and which belong to both of those syllables. For example, the first `[s]␣`in the transcription *[ap][bla[s]@n]* of *abblassen* is part of the second syllable and the third syllable, whereas the `[z]␣`in the transcription *[ap][bla:][z@n]* of *abblasen* belongs to the third syllable only.

The first syllabified wordform transcription column uses the SAM-PA character set, and syllable boundaries within words are shown by hyphens. The FLEX name and description of this column are as follows:

PhonSylSAM `Syllabified␣phonetic␣wordform,␣SAM-PA␣character␣set␣`

The next two columns both use the CELEX character set. The first marks every syllable boundary within each transcription with a hyphen. The FLEX name and description of this column are as follows:

PhonSylCLX `Syllabified␣phonetic␣wordform,␣CELEX␣character␣set␣`

The other CELEX syllabified phonetic wordform column uses the brackets notation as described above, and its FLEX name and description are as follows:

PhonSylBCLX `Syllabified␣phonetic␣wordform,␣CELEX␣character␣set␣(brackets)␣`

The next column gives syllabified wordform transcriptions in the CPA character set. Every syllable boundary within each word is marked by a full stop. The FLEX name and description of this column are as follows:

PhonSylCPA `Syllabified␣phonetic␣wordform,␣CPA␣character␣set␣`

The fifth column uses the DISC character set, and here every syllable boundary within each word is marked by a hyphen. The FLEX name and description of this column are as follows:

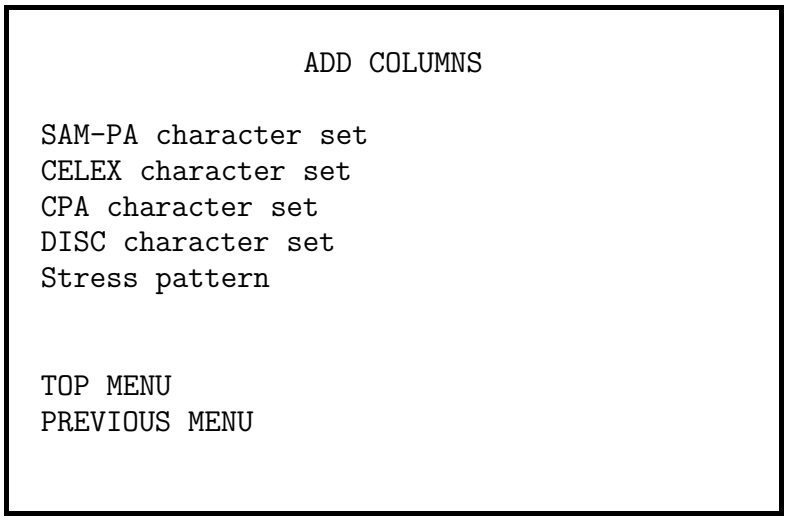
PhonSylDISC `Syllabified␣phonetic␣wordform,␣DISC␣character␣set␣`

The last column in this subsection gives counts of the phonetic syllables which occur in each transcription. For example, both *abblasen* and *abblassen* contain 3 syllables. The FLEX name and description of this column are as follows:

SylCnt `Wordform,␣number␣of␣phonetic␣syllables␣`

2.1.2.3 TRANSCRIPTIONS FOR STRESSED AND SYLLABIFIED WORDFORMS

This set of columns gives syllabified transcriptions that also mark the points of primary stress in each wordform. These are the columns you can choose from:



The first column uses the SAM-PA character set, and as well as using hyphens to mark syllable boundaries, these transcriptions show points of primary stress by means of the ‘ double quote’ character ("). This character is placed immediately before a stressed syllable. The FLEX name and description of this column are as follows:

PhonStrsSAM `Syllabified␣phonetic␣wordform,␣with␣stress␣marker,␣SAM-PA␣character␣set␣`

The second column uses the CELEX character set, and as well as using hyphens to mark syllable boundaries, these transcriptions show the points of primary stress with an inverted comma (’) immediately before the stressed syllable.

The FLEX name and description of this column are as follows:

PhonStrsCLX `Syllabified␣phonetic␣wordform,␣with␣stress␣marker,␣CELEX␣character␣set␣`

The third column uses the CPA character set, including full stops to mark syllable boundaries, and these transcriptions show points of primary stress with an inverted comma (’) immediately before the stressed syllable. The FLEX name and description of this column are as follows:

PhonStrsCPA `Syllabified␣phonetic␣wordform,␣with␣stress␣marker,␣CPA␣character␣set␣`

The fourth column uses the DISC character set, and along with hyphens to mark syllable boundaries, these transcriptions show points of primary stress with an inverted comma (') immediately before the stressed syllable. The FLEX name and description of this column are as follows:

PhonStrsDISC `Syllabified␣phonetic␣wordform,␣with␣stress␣marker,␣DISC␣character␣set␣`

The last column in this subsection contains a simple stress pattern for each wordform. A *stress pattern* is a string which shows how each phonetic syllable is stressed in speech. Each syllable is represented by one numeric character: either `0␣` or `1`. `1␣`indicates that the syllable receives primary stress, and `0␣`that it does not receive primary stress. Thus the four-syllable word *Biologe* has the stress pattern `0010␣`and *Biologie* has the pattern `0001`. Note that patterns with more than one 1 can occur. The FLEX name and description of this column are as follows:

StrsPat `Wordform,␣stress␣pattern␣`

2.2 PHONETIC PATTERNS

Phonetic patterns here means CV patterns: the consonant and vowel patterns for the phonetic transcription (as opposed to the orthographic or phonological transcriptions) of any lemma (headword or stem) or wordform you select. Instead of the basic CV pattern, which uses hyphens to mark phonetic syllable boundaries within words, you may want to use the alternative notation which delimits syllables by means of square brackets. The phonetic CV pattern used here represents each *short vowel* as `V`, each *long vowel* and *diphthong* as `VV`, and each *consonant* and *affricate* as `C`. In addition, special consideration is made for *ambisyllabic* consonants, such as the `[s]␣`in the word *abblassen*. (Ambisyllabic consonants are those consonants which seem to 'belong' to two syllables at once.) The `[s]␣`is replaced by one `C␣`at the end of the first syllable, and *another* `C␣`at the beginning of the second syllable. Thus its CV pattern is `VC-CCVC-CVC`. With a brackets notation, the ambisyllabic nature of the consonant can be made clearer: `[VC][CCV[C]VC]`.

This table illustrates the two different formats you can choose for you CV patterns:

		CV pattern	CV pattern with brackets
abblasen	`[ap-bla:-z@n]`	`VC-CCVV-CVC`	`[VC][CCVV][CVC]`
abblassen	`[ap-bla-s@n]`	`VC-CCVC-CVC`	`[VC][CCV[C]VC]`

2.2.1 PHONETIC CV PATTERNS FOR HEADWORDS

For headwords, the basic phonetic CV patterns include hyphens as syllable markers. The FLEX name and description of this column are as follows:

PhonCV ***(PhonCVLemma)*** `Headword,␣phonetic␣CV␣pattern␣`

Alternatively you can choose phonetic CV patterns of headwords which use square brackets to delimit the syllables. This column has the following FLEX name and description:

PhonCVBr ***(PhonCVBrLemma)*** `Headword,␣phonetic␣CV␣pattern,␣with␣brackets␣`

2.2.2 PHONETIC CV PATTERNS FOR STEMS

For stems, the basic CV pattern with hyphens as syllable markers are given in the column whose FLEX name and description are as follows:

PhonStCV ***(PhonStCVLemma)*** `Stem,␣phonetic␣CV␣pattern␣`

The other column with phonetic CV patterns for stems includes square brackets to delimit syllables. Its FLEX name and description are as follows:

PhonStCVBr ***(PhonStCVBrLemma)*** `Stem,␣phonetic␣CV␣pattern,␣with␣brackets␣`

2.2.3 PHONETIC CV PATTERNS FOR WORDFORMS

Two phonetic CV pattern columns are available for wordforms. The first uses hyphens to mark syllable boundaries within wordforms, and its FLEX name and description are as follows:

PhonCV `Wordform,␣phonetic␣CV␣pattern␣`

The second uses square brackets to delimit the syllables in each wordform. Its FLEX name and description are as follows:

PhonCVBr `Wordform,␣phonetic␣CV␣pattern,␣with␣brackets␣`

2.3 PHONOLOGICAL TRANSCRIPTIONS FOR STEMS

The phonological representations provided have been automatically generated using the available CELEX phonological and morphological information. They are available only for the stem form of certain lemmas. Not all stems have phonological representations, but only those with enough information, both phonological and morphological, to make the automatic formation of a transcription possible. The transcriptions given are not necessarily the definitive underlying forms in the strict linguistic sense, though they are certainly abstract (they leave out the information which can be formulated by applying certain phonetic rules to them).

Every transcription gives a phonological representation of each *morpheme* in the stem. When the word consists of more than one morpheme, the boundary between two morphemes is marked in one of two ways: either *type 1* (shown by the symbol `+`) or *type 2* (shown by the symbol `#`).

A *type 1* morpheme boundary means (amongst other things) that when the two elements are joined, the morpheme boundary given normally does *not* coincide with the phonetic syllable boundary. Such boundaries usually occur between a stem and a suffix – the transcription for *Arbeiter* (i.e. the stem *Arbeit* plus the affix *-er*) is `arbait+@r␣`(CELEX character set).

A *type 2* morpheme boundary means (amongst other things) that when the two elements are joined, the morpheme boundary given often does coincide with the syllable boundary. Such boundaries usually occur between prefixes and stems, or between two stems – the transcription for *Arbeitgeber* (i.e. the stem *Arbeit* plus the stem *Geber* is `arbait#ge:b+@r␣` (CELEX character set).

The provision of these two distinct types of morpheme boundary is helpful when you want to investigate rules which govern sound changes in complex words. Each morpheme is given in its original 'underlying' (i.e. a phonological not phonetic) state. The complex word *Arbeitgeber* thus has as its transcription `arbait#ge:b+@r`, where the underlying phonological form of the stem *geb* is `ge:b`. Table 4 below sets out the phonological and phonetic transcriptions of the examples so far discussed (plus a few extra) to illustrate the difference between phonological transcriptions and phonetic syllabified transcriptions.

Stem	Phonological Transcription	Phonetic Transcription
Arbeiter	`arbait+@r`	`[ar][bai][t@r]`
Arbeitsplatz	`arbait+s#plats`	`[ar][baits][plats]`
Arbeitgeber	`arbait#ge:b+@r`	`[ar][bait][ge:][b@r]`
arbeitsamkeit	`arbait#za:m#kait`	`[ar][bait][za:m][kait]`

Table 4: Phonological vs. phonetic transcriptions

Counting the total number of phonological transcriptions shows that not every stem in the database has such a transcription. There are two reasons why a stem may not be accompanied by a phonological transcription. First, there may not be enough morphological information available to give a full analysis of a particular word. (The German morphological stem column ***Status*** indicates whether or not a complete analysis is available.) Second, there may not be enough phonological information to give a complete transcription. The absence of information for one morpheme in a particular word means that no transcription can be given. Compounds which include abbreviations or proper nouns, for example, thus have no phonological transcriptions.

Also, you should note that because phonological representations have been derived from the 'deepest' segmentation

available (i.e. from the Flat Segmentation, involving only simple free and bound morphemes), these transcriptions may radically differ from corresponding phonetic transcriptions. Thus a word like *Bodenfrost* emerges through processes of stem allomorphy with a phonological transcription `[bo:d@n#fri:r]`.

Finally, it should be emphasized that you are dealing here with automatically-generated information; detailed correction by knowledgeable humans has not been carried out. In general, though, these tentative transcriptions are correct so long as the word is regular.

You can choose transcriptions in the CELEX or SAM-PA phonetic character coding sets (see table 2 in section 2.0.1 above). Phonological transcriptions are not available in DISC, however, since that coding set uses the boundary marker codes (`#␣`and `+`) as character codes in their own right. You should note that phonological representations are available only for stems, not headwords or wordforms. Phonological transcriptions are thus available in lemma lexicons, and the names of these columns are the first of the two names given in the margin with each definition. There are no phonological transcriptions for wordforms, but you can see the phonological information for each wordform's stem by using the lemma information given with the morphology columns for German wordforms. The names of these columns are the ones given in brackets directly underneath the lemma lexicon names.

First, the FLEX name and description of the column which gives phonological transcriptions in the SAM-PA character set:

PhonolSAM `Phonological␣deep␣structure,␣SAM-PA␣character␣set␣`
(PhonolSAMLemma)

And second, the FLEX name and description of the column which gives phonological transcriptions in the CELEX character set:

PhonolCLX `Phonological␣deep␣structure,␣CELEX␣character␣set␣`
(PhonolCLXLemma)

3 GERMAN MORPHOLOGY

Morphological information for German is available with lemma lexicons and wordform lexicons. If you are interested in inflectional morphology, then you should use a wordforms lexicon, and if you are interested in derivational and compositional morphology, you should use a lemma lexicon.

3.1 MORPHOLOGY OF GERMAN LEMMAS

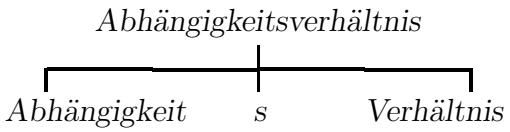
The morphological analyses given for lemmas in the CELEX databases always use the *stem* form of the lemma, because this form is usually the shortest in any inflectional paradigm, without any visible inflectional endings. Before finding out details about each of the columns available, you should look at the sections below which try to give some explanation of the methods used to obtain the analyses given in the database. You will then know what CELEX means by terms such as *immediate segmentation, hierarchical segmentation, compound, derivation*, and *derivational compound*. You will also know how CELEX treats the special 'problem' compound cases which can be treated as derivational compounds *and* ordinary compounds. After all that, you'll understand more clearly what each of the various columns has to offer.

3.1.1 HOW TO SEGMENT A STEM

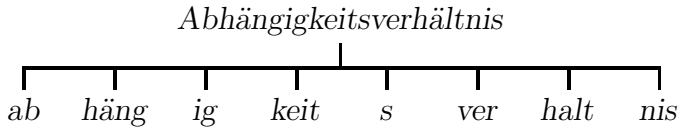
The first and most fundamental type of segmentation is *immediate segmentation*. This simply involves splitting a stem into its largest constituent parts. If you continue to carry out immediate segmentation until there is nothing left to segment, you arrive at the stem's *complete segmentation*. Depending on your requirements, you can look at a complete segmentation in two forms. The first is the *flat* form, which shows every morpheme that makes up the stem. The second is the *hierarchical* form, which, as well as pointing out the individual morphemes in a stem, also shows all the analyses which have to be made to identify those morphemes. The flat segmentation gives the conclusion reached; the hierarchical segmentation shows the working.

To illustrate the three types of segmentation, take as an example the word *Abhängigkeitsverhältnis*.

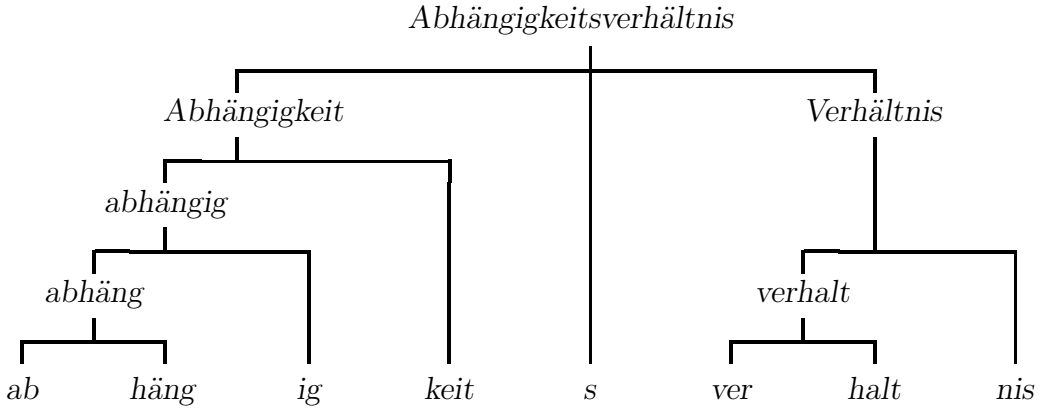
The first type of analysis ' immediate segmentation' gives the stem *Abhängigkeit* plus the affix ('link morpheme') -*s*- plus the stem *Verhältnis*:



The second type of analysis 'complete segmentation (flat)' shows you what you get if you keep applying immediate segmentation, namely the constituent morphemes of *Abhängigkeitsverhältnis*: the affix *ab* plus the stem *häng* plus the affix *ig* plus the affix *keit* plus the affix ('link morpheme') *s* plus the affix *ver* plus the stem *halt* plus the affix *nis*.



The third type 'complete segmentation (hierarchical)' shows you the full analysis of the word, including each individual immediate segmentation carried out. It gives you enough information to produce a hierarchical tree diagram like this one:



For most stems in the database, representations of each of these three types of segmentation are available. Sometimes there is more than one representation, because certain stems can have more than one immediate segmentation. To explain this fully, the next section describes the basic analyses that result from immediate segmentation.

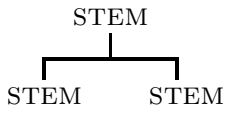
3.1.2 HOW TO ASSIGN AN ANALYSIS

When you attempt to split a stem into its biggest component parts, the result is always some combination of *stems* plus *affixes*. The most straightforward case of all is a stem which consists of only one (free) morpheme: it is *monomorphemic*, and clearly can't be split up. Every other stem, however, consists of one smaller stem plus at least one affix or one other stem, and can be termed either a *Compound*, or a *Derivation*, or a *Derivational Compound*. It is important to understand the differences between these three terms, since they are at the heart of the morphological information CELEX provides. So, in the subsections below, each is defined in terms of stems and affixes. Examples are given, and simple 'tree' diagrams illustrate the appropriate immediate analyses.

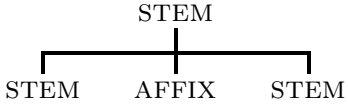
3.1.2.1 THE COMPOUND

A COMPOUND is the joining of two stems into one new stem. The immediate analysis always takes one of two forms:

(i) a binary split into two stems (the word *Haustür* for example: *Haus + Tür*).



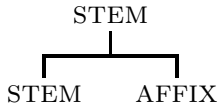
(ii) a triform split into a stem, an affix (simply a 'link' morpheme), and a stem (the word *Badewanne* for example: *Bad + e + Wanne*).



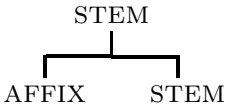
3.1.2.2 THE DERIVATION

A DERIVATION involves affixation, whereby affixes can be added to an existing stem to form a new stem. The immediate analysis always takes one of four possible forms:

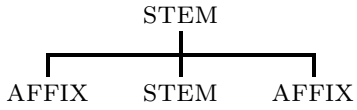
(i) a binary split into a stem and an affix (the word *Fehlerhaft*, for example: *Fehler + haft*).



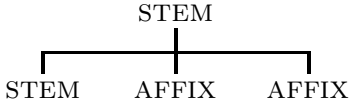
(ii) a binary split into an affix and a stem (the word *Mißklang* for example: *miß+ Klang*).



(iii) a triform split into an affix, a stem, and an affix (the word *Gerede* for example: *ge + red + e*).



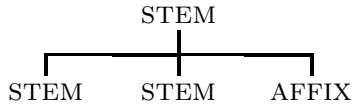
(iv) a triform split into a stem, an affix, and an affix (the word *anspruchslos* for example: *Anspruch + s + los*).



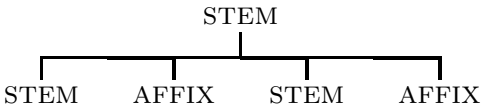
3.1.2.3 THE DERIVATIONAL COMPOUND

A DERIVATIONAL COMPOUND is a compound which can only be formed in combination with a derivational affix (as opposed to a simple link morpheme). The immediate analysis always takes one of two forms:

(i) a triform split into a stem, a stem, and an affix (the word *achtkantig* for example: *acht + Kante + ig*).

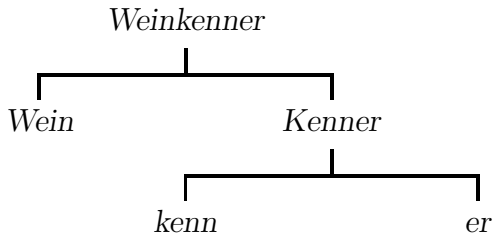


(ii) a quaternary split into a stem, an affix, a stem, and an affix (the word *achtzigjährig* for example: *acht + zig + Jahr + ig*).

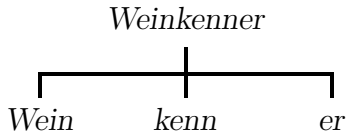


3.1.2.4 COMPOUND OR DERIVATIONAL COMPOUND?

The general definition of a derivational compound is normally sufficient, but when the second stem is a verbal form, things become more complicated. A stem which comprises a noun plus a verb plus an affix can normally be considered a derivational compound, but some people may want to treat it as an ordinary compound. The distinction is important, since it can affect not only the appearance of a single immediate segmentation branch, but also the appearance of a complete hierarchical tree. The stem *Weinkenner* is such a 'problem' compound. If you consider it to be an ordinary compound (the stem *Wein* plus the stem *Kenner)*, its complete hierarchical tree looks like this:



But if you consider it to be a derivational compound, the first immediate segmentation gives you the stem *Wein* plus the stem *kenn* plus the affix *er*, which gives the full hierarchical tree a different appearance:



So, when you're faced with a compound that includes a verbal component and an affix, how do you decide whether it's an ordinary compound, a derivational compound, or both? To illustrate the principles used in analysing the information to you, consider the computer program-like algorithms set out below. They take as their initial premise that the word you are looking at can be analysed as a noun, an adverb, an adjective, or a preposition plus a verb and an affix. As the algorithms show, just because they *can* be analysed this way, it is not always true that they *should* be analysed this way. When you come to select columns containing morphological analyses from the database, you can choose for yourself the analysis you want to see. Figuring out these algorithms now will help you to understand the options you can choose from.

First, here are the variables used in the algorithms and their definition:

n	is a noun
v	is a verb
a	is an adjective or an adverb
$prep$	is a preposition
aff	is an affix

$[n + v + aff]$

if n is the direct object of v
then if $[n + v + aff]$ is a specific sort of $v + aff$
then $[n + v + aff]$ is a COMPOUND
and a DERIVATIONAL COMPOUND
else $[n + v + aff]$ is a DERIVATIONAL COMPOUND
else $[n + v + aff]$ is a COMPOUND $[n + n]$

How do these rules apply in practice? Take as an example the word *Radfahrer*. The first question is whether the noun *Rad* is the direct object of the verb *fahren*. The answer is yes, so move to the 'then' clause for the next question: is *Radfahrer* a specific sort of *Fahrer*? Again, the answer is yes, so on moving to the next 'then' clause, you get the answer that *Radfahrer* is one of those words which can be treated as an ordinary compound *and* as a derivational compound. Its immediate analysis can be noun plus noun *(Rad + Fahrer)* or, as originally suspected, noun plus verb plus affix *(Rad + fahr + er)*. In such cases, the CELEX database offers you both analyses of the stem. Using the 'status of analysis' columns, your lexicon can include either sort of analysis or both of them, according to your preference.

Another example: *Säbelrassler*. The first question is whether the noun *Säbel* is the direct object of the verb *rasseln*. The answer is yes, so move to the 'then' clause for the next question: is *Säbelrassler* a specific sort of *Rassler*? Here the answer has to be no, since the word *Rassler* does not exist by itself. So, move to the 'else' clause to discover that *Säbelrassler* can only be a derivational compound. Its immediate analysis is thus noun plus verb plus affix: *Säbel + rassel + er*.

One last example: *Gewohnheitstrinker*. The first question is whether the noun *Gewohnheit* is the direct object of the verb *trink*. The answer this time is quite clearly no, so move straight to the last 'else' for the answer: *Gewohnheitstrinker* is just an ordinary compound with the simple binary split into a noun plus a noun: *Gewohnheit + s + trinker* (in this case with an extra link morpheme 's')

There is also a simple algorithm for stems which can be

analysed as adjective or adverb plus verb plus affix:

$[a + v + aff]$

if $[a + v + aff]$ is a specific sort of $[v + aff]$
and if $[a + v + aff]$ means the same as $[(det)\ a\ n]$
then $[a + v + aff]$ is a COMPOUND $[a + n]$
else $[a + v + aff]$ is a DERIVATIONAL COMPOUND

This time there are two questions which have to be answered together. If one answer, or neither answer, is positive, then the stem is a derivational compound. If both answers are positive, then the stem is an ordinary compound. Thus with the stem *Schwerarbeiter*, the first question is whether it is a particular type of *Arbeiter*—and the answer is yes. The second question is whether *Schwerarbeiter* means the same as *(ein) schwerer Arbeiter*—and the answer is no. So, since one of the two answers is negative, you must go to the 'else' clause. This tells you that the stem is a derivational compound.

In fact, most adjective-or-adverb-plus-verb-plus-affix stems are derivational compounds; you won't often find a stem that produces a positive answer to both the questions.

Another important category to consider here is the preposition plus verb plus affix combination. Usually, they can be analysed simply as verb plus affix, i.e. as simple derivations. However on occasions such stems can better be analysed as derivational compounds. The algorithm below indicates when:

$[prep + v + aff]$

if $[prep + v]$ is an existing verbal stem with
the equivalent meaning
then $[prep + v + aff]$ is a DERIVATION $[v + aff]$
else $[prep + v + aff]$ is a DERIVATIONAL COMPOUND

Take as an example the word *Ausbrecher*. The question is whether the verb *ausbrech* is a verb that exists in its own right, and the answer is yes. Naturally this analysis takes account of the meaning of the word – if *Ausbrecher* did not mean *jemand der ausbricht* then clearly the analysis would be wrong. So, the answer yes lets you move onto the 'then' clause, where you find out that the stem is in fact a derivation with an immediate two-part analysis of verb plus affix.

Another example is the word *Umwohner*. Here the verb *umwohnen* does not exist, so the 'else' option indicates that this word is a derivational compound with a triform immediate analysis of preposition plus verb plus affix.

These detailed definitions and explanations are given so you know what to expect when you ask for morphological analyses of stems. You can control the number of analyses you see for each stem, as well as the type of analyses, by means of restrictions on the 'number' and 'status' columns which are defined below. You can decide for yourself whether your lexicon should contain just one 'default' analysis per stem, or whether it should contain more than one analysis per stem. In cases where a stem can be analysed as a compound or a derivational compound, you can choose in theory to include whichever type you prefer, leaving out the other type. In short, you have the freedom to build lexicons which contain morphological information in the form you most prefer.

Having set out much of the theory behind the morphological analyses provided by CELEX, it's now possible to discuss the columns themselves, and this is done in the sections which follow.

3.1.3 STATUS AND SEPARABLE

The first ADD␣COLUMNS␣menu you see after you select the 'Morphology' option is this one:

```
                ADD COLUMNS

Status
Derivational/compositional information   >
Separable
Inflectional paradigm
Inflectional variation


TOP MENU
PREVIOUS MENU
```

Before dealing with the various derivational/compositional information columns, which form the bulk of the available morphological information, the first column and the third column can be quickly dealt with here.

The first column simply tells you by means of a single code whether each stem is morphologically simple, morphologically complex, or why it is as yet unanalysed. These are the codes that are used:

Status	Code	Example
Morphological analysis available:		
Morphologically complex	C	*Abendessen*
Conversion (zero derivation)	Z	*Abflug*
Monomorphemic	M	*Abend*
Morphological analysis unavailable:		
Morphology irrelevant	I	*Abakus*
Lexicalised flection	F	*anhaltend*
Morphology undetermined	U	*Adamit*

Table 5: Derivational morphology status codes

If a stem contains at least one stem plus at least one other stem or affix, then it is said to be morphologically complex. Details of how the stem can be analysed are given in the derivational/compositional segmentation columns described in the section below. Thus if a stem has the morphological status code C␣ for 'complex', you know that information about its derivational and/or compositional morphology is available in the database.

If a stem is monomorphemic, then it contains only one morpheme, and no further analysis is required. The morphological status code M␣means 'monomorphemic', and you know that a simple one-stem analysis is given as the derivational and/or compositional morphology for each stem with this code.

If a stem appears to be derived from another stem which is identical in form but different in word class, it gets the code Z␣for 'zero derivation' or conversion. The noun *Abfall*, for example, can be said to derive from the verb *abfallen*. Normally derivations from one word class to another are clearly marked by means of an affix – *kegeln* is a verb derived from the noun *Kegel*, for example. But conversions, on the other hand, are not so marked: it's as if an affix containing nothing had been added to the original stem. In some cases, however, the process of conversion causes changes in the central vowel of the stem. This phenomenon, called allomorphy, is dealt with below.

Sometimes morphological analysis is not appropriate for a particular stem. Usually this is true when the stem involves a proper noun in some way (*Achensee*, for example), or when the stem has an extended or sentence-like structure (such as the phrase *Aufundabgehen*), or when the stem is an interjection (for example *ach*). Thus when a stem has the code `I␣` for ' irrelevant', you know that a morphological analysis isn't considered necessary, and that its entries in the segmentation columns described below are therefore empty.

On occasions, a particular flectional form of a stem occurs very frequently, or acquires a meaning slightly different from that of the original stem. For this reason, they can be given stem status in their own right, rather than being considered mere flections. Typically, present and past participles become independent adjectives. In the *Brockhaus-Wahrig Deutsches Wörterbuch*, the word *abgelebt* is listed as a bold-type entry in its own right as well as a flection of the verb *ableben*. Forms such as these are called *lexicalised flections*. For the CELEX database, any such word which appears as a bold-type headword in the *Brockhaus-Wahrig Deutsches Wörterbuch* is given the morphological status code `F␣`for 'flection'. The morphological properties of such words are given with the inflectional information available in the 'Morphology of German wordforms' columns. For this reason, no analyses are given for them with the compositional and derivational information.

The last of the morphological status codes is the one which covers everything else. It simply means that the stems in question couldn't be satisfactorily analysed, for a variety of reasons. Some stems use classical affixes, which don't behave quite like normal German affixes (*Aerogramm* for example), other stems are recent foreign loanwords which aren't always normal productive German stems (as in *Rembours*), and others are just plain weird (as in *Wirrwarr*). In all such cases the morphological status code is `U␣`for 'undetermined', and no analyses are given.

This column can be used to eliminate from your lexicon stems for which there are no morphological analyses, allowing you to concentrate on those which do. Simply add a restriction which states that you only want stems which are morphologically complex: `MorphStatus␣=␣C`.

The column which contains these morphological status codes has the following FLEX name and description:

MorphStatus `Morphological␣status␣`
(MorphStatusLemma)

The third option deals with *separable* stems: those stems—mostly verbs—whose wordforms sometimes split into two parts, depending on the structure of the sentence they are used in. The stem *auspack*, for example, is the same stem whether it occurs in a phrase like *Wenn er das tut dann packe ich aber mal aus* or in a phrase like *Ich will zuerst den Koffer auspacken*. So, if any wordforms of a stem can occur in this way, this column includes the code `Y`. If not, the code given is `N`. This column can be used in the construction of a restriction which specifically includes such stems in your lexicon or specifically excludes them from your lexicon. The FLEX name and description of this column are as follows:

Sepa `Separable␣`
(SepaLemma)

3.2 INFLECTIONAL PARADIGM

The fourth option deals with the *inflectional paradigm* of stems. Each stem in the database receives one of the codes shown in table 2.

CODE	MEANING
A	Adjectival inflection for noun
I	Inflected but no paradigm available
U	Uninflected
i...	Irregular verb
r1	Standard verb
r2	Regular verb ending in “d/t” or “(plosive/fricative)+(m/n)”
r3	Regular verb ending in “schwa+r”
r4	Regular verb ending in “schwa+l”
r5	Regular verb ending in “vowel” or “vowel+h”
r6	Regular verb ending in sibilant
S...	Singular nominal flection
P...	Plural nominal flection

Table 6: Inflectional paradigm codes

The numerical noun codex are described in the *Appendices*, Table of flections of German nouns. The codes used in this column should be interpreted in the following way:

Let's take as an example the word *Auto* which is a noun with the inflectional features *S1* and *P5*. The code *S1* means that an *s* is added to this noun if the genitive form *des Autos* is used and all other flections of this noun in its singular form appear as *Auto*. The code *P5* means that the word *Auto* will receive an *s* in all four plural flections. For every noun the 'S' and 'P' codes appear concatenated by a slash, as for *Birne*, which has been assigned the code *S3/P3*.

A *u* added to the codes for the plural flections means that the plural flections of this noun will receive an "Umlaut" on the vowel of the stem.

There are two codes that may cause some confusion, i.e. *S0* and *P0*. *S0* means that we are dealing with a noun that can only be used in its plural form, whereas a noun with the code *P0* can only be used in its singular form.

The alphanumeric verb codes have been derived from the conjugation tags found in the *Brockhaus-Wahrig Deutsches Wörterbuch* (1980, pp. 21 - 25). A description of these codes can be found in the *Appendices* Table of Conjugations of German Verbs. The codes used in this column should be interpreted in the following way:

The verb *verhelfen* is a verb with code *i165*. This means that the inflectional paradigm of this verb is the same as the verb *helfen*, which is mentioned in the Table of Conjugations of German Verbs as the example for verbs with code *i165*. The FLEX name and description of this column are as follows:

InflPar `Inflectional␣paradigm␣`
(InflParLemma)

3.3 INFLECTIONAL VARIATION

It is sometimes possible that there is more than one alternative for the inflectional paradigm of a noun. For example the word *Ding* can have two different plural forms, i.e. *Dinger* and *Dinge*. In this case there will appear a 'Y' in the Yes/No column ***Inflectional variation***, which means that there are

more paradigms for either the singular forms or the plural forms of this noun. In the ***InflPar*** column, we only listed the first alternative, which has to be regarded as the main variant. The decision for choosing between the alternatives is mainly based on *Duden Rechtschreibung* and on *Brockhaus-Wahrig Deutsches Wörterbuch*. The result of this decision is that a word like 'Abbau' is coded as 'S1/P1' which means that this word receives an 's' in the genitive singular form and an 'e(n)' ending for the plural forms. However the plural form 'Abbauten' is allowed as well. This means that the code for plural forms can also be 'P10'. As stated before no secondary or even tertiary forms are included. The fact that there is an other paradigm can be derived from the fact that this column states: "Yes there is an other paradigm". The FLEX name and description of this column are as follows:

InflVar `Inflectional␣variation␣`
(InflVarLemma)

3.4 DERIVATIONAL/COMPOSITIONAL INFORMATION

```
                ADD COLUMNS

Number of morphological analyses
Morphological analysis number (0-N)
Status of morphological analysis          >
Segmentations                             >
Other                                     >


 TOP MENU
 PREVIOUS MENU
```

These options give you information about the derivational and compositional morphology of *stems*, including how many analyses are available for each stem, a unique number for each analysis, an indication of the way in which each analysis has been made, and a marker for the 'default' analyses for each stem.

The first option is a column which simply indicates how many analyses have been made for each stem. For example, *Abendessen* has one analysis, *Abbaufeld* has two. The

number of analyses for each stem also equals the number of rows that stem can have with distinct analyses, since each morphological analysis is assigned to its own individual row.

You can use this column to construct restrictions for your lexicon. A simple example would be one that includes in your lexicon only those stems which have more than one analysis. This would take the form `MorphCnt␣>␣1`. The FLEX name and description of this column are as follows:

MorphCnt ***(MorphCntLemma)*** `Number␣of␣morphological␣analyses␣`

The second option is a column which identifies each analysis of a particular stem. Each different morphological analysis of a stem is assigned to a different row, and this column gives the number of the row. Thus the lemma *Abbaufeld* has two rows: one has the ***MorphNum*** 1, the other has the ***MorphNum*** 2. The FLEX name and description of this column are as follows:

MorphNum ***(MorphNumLemma)*** `Morphological␣analysis␣number␣(0-N)␣`

3.5 STATUS OF MORPHOLOGICAL ANALYSIS

Under the 'status of morphological analysis' option there are three 'yes/no'-type columns which, when you use them to construct restrictions, can help you extract the analyses you want from the many stem segmentations available.

Each distinct morphological analysis of each stem has a number, and is given (in several different forms) on its own row in the database. These columns give simple information about each analysis, and are particularly useful whenever a stem is a 'problem' compound, or whenever it contains a 'problem' compound. (A problem compound, as discussed in section 3.1.2.4, can correctly be analysed as a derivational compound or an ordinary compound.) The three columns in question are called ***DerComp, Comp***, and ***Def***.

Whenever ***DerComp*** contains a `Y`, you know that 'yes, any problem compounds which occur anywhere in this stem are analysed as derivational compounds'. And naturally, `N␣`

means that problem compounds *aren't* analysed as derivational compounds.

DerComp `Derivational␣compound␣analysis␣method␣`
(DerCompLemma)

Whenever ***Comp*** contains a `Y`, you know that 'yes, any problem compounds which occur anywhere in this stem are analysed as ordinary compounds'. And again, `N␣`means that any problem compounds *aren't* analysed as ordinary compounds.

Comp `Compound␣analysis␣method␣`
(CompLemma)

Whenever ***Def*** contains a `Y`, you know that 'yes, this analysis is the default analysis'. If a stem includes a problem compound, then there are *two* default analyses with a `Y␣`in this column, one with the derivational compound type analysis, the other with the ordinary compound type analysis.

Def `Default␣analysis␣`
(DefLemma)

To illustrate how you can use these columns, imagine that you have chosen ***Imm*** as the form of morphological analysis you want to see (this column, and the other columns containing the same analysis in different forms, are described in the sections following this one). Then say that you are interested in the stem *Absichtserklärung*, which has two different analyses. It is one of the problem compounds which can be a derivational compound or an ordinary compound, which accounts for two analyses.

First you can decide whether you want just one default analysis, or whether you want to see both available analyses.

If you want to see its possible segmentations, then you don't need to add extra restrictions. As the ***MorphCnt*** column indicates, there are 2 analyses given for this stem, *Absichtserklärung*, so this is what the unrestricted example lexicon looks like:

Stem	MorphNum	DerComp	Comp	Def	Imm
Absichtserklaerung	1	Y	N	Y	Absicht+s+erklaer+ung
Absichtserklaerung	2	N	Y	Y	Absicht+s+Erklaerung

Analysis number 1 is a derivational compound, so in this case ***DerComp*** contains `Y`, and ***Comp*** contains `N`. Analysis number 2 is an ordinary compound, so there ***Comp*** contains `Y`, and ***DerComp*** contains `N`.

However, rather than including both forms in your lexicon, you might want to ignore the ordinary compound analysis, and just see the derivational compound analysis. To do this for all the stems in the database, you should add an 'expression' restriction to your lexicon which states that `DerComp␣=␣Y`. In the example lexicon, this one restriction produces the following result:

Stem	MorphNum	DerComp	Comp	Def	Imm
Absichtserklaerung	1	Y	N	Y	Absicht+s+erklaer+ung

In the same way, if you want to ignore the derivational compound analyses in favour of the ordinary compound analyses, you should add an 'expression' restriction to your lexicon which states that `Comp␣=␣Y`. In the example lexicon, this restriction produces the following result:

Stem	MorphNum	DerComp	Comp	Def	Imm
Absichtserklaerung	2	N	Y	Y	Absicht+s+Erklaerung

Rather than seeing a number of analyses, you might prefer to look at just one straightforward default analysis, no matter how many alternatives are given in subsequent rows. Again, you can quickly construct restrictions to make this possible. The quickest way is to use the ***MorphNum*** column, which gives a number to each analysis of each stem. You can say `MorphNum␣=␣1`, which means that only the very first analysis of each stem appears in your lexicon. And whenever a stem is a problem compound, you should remember that the first analysis is always the derivational compound form rather than the ordinary compound form.

Another way to get a single analysis for each stem with problem compounds treated as derivational compounds is to add these two restrictions: `Def␣=␣Y␣`and `DerComp␣=␣Y`. Here you are saying explicitly that you want the default form of the stem (in the example lexicon that means ignoring the '*Erklärung* is a noun' analysis) and that whenever problem compounds occur, you want to see the derivational compound form.

Whether you choose the single ***MorphNum*** restriction or the two ***Def*** and ***DerComp*** restrictions, the effects on your lexicon are the same. The resulting example lexicon looks like this:

Stem	MorphNum	DerComp	Comp	Def	Imm
Absichtserklaerung	1	Y	N	Y	Absicht+s+erklaer+ung

If you want one analysis, and if in the case of problem compounds you want that one analysis to be an ordinary compound rather than a derivational compound, all you have to do is add two restrictions. First, ask for a default analysis by saying `Def␣=␣Y`; this omits the non-preferred analyses like the '*erklär* is a verb' option. Then specify that you want any problem compounds to be given as ordinary compounds by adding the restriction `Comp␣=␣Y`. This is what the example lexicon then looks like:

Stem	MorphNum	DerComp	Comp	Def	Imm
Absichtserklaerung	2	N	Y	Y	Absicht+s+Erklaerung

These explanations may appear complicated, but by reading them, you can get to know the important restrictions that you can use to extract the types of analysis you really want.

3.5.1 IMMEDIATE SEGMENTATION

Immediate segmentation is the least detailed form of analysis offered here. It doesn't give you a full analysis, right down to all the smallest elements a stem contains; rather it is a simple, one-level breakdown of a stem into its next biggest elements. So, while complete segmentation is equivalent to a full analytical tree, immediate analysis can be thought of as a close look at a particular level.

There are six columns which present the immediate segmentation of stems to you. The first gives the orthography of the analysed elements. The next two give more general coding, so that using the FLEX options `SHOW␣`and `QUERY`, you can look for stems which have a particular form: a preposition plus a noun, say, or a stem plus a stem plus an affix. The last three indicate whether stem allomorphy, vowel mutation (Umlaut) or a change of meaning (Opacity) occurs in the immediate analysis of a stem.

In the first column, you get the orthography of the first-level elements themselves, each separated by a `+␣`sign. Diacritical markers are not included. Thus the stem *Inhaber* is shown as `in+hab+er␣`, in accordance with the various rules discussed in section 3.1.2.4. Note that each element is given in the form of a stem or an affix, even when the original word doesn't use that particular form. Thus the stem *achtkantig* is analysed as `acht+Kante+ig`, where *kant* is re-written in the form of the stem *Kante*. The FLEX name and description of this column are as follows:

Imm `Immediate␣segmentation␣`
(ImmLemma)

The second column is like the first, except that where the first column gives you the orthography of each element, this column gives you the word class of each element.

Word Class	Label
Adjective	`A`
Adverb	`B`
Conjunction	`C`
Article	`D`
Interjection	`I`
Noun	`N`
Pronoun	`O`
Preposition	`P`
Quantifier/Numeral	`Q`
Verb	`V`
Abbreviation	`X`
Affix	`x`
Contracted Preposition	`c`
Lexicalized Flection	`F`
Node	`n`
Preposition as part of a node	`p`
Root	`R`

Table 7: Word class labels (immediate segmentation)

Single letter labels are used to represent the syntactic class of each element – which is unlike many of the syntactic codes used in other parts of the database. The use of a single character means that there is no possibility of a code becoming ambiguous, since each character is unique. The previous table shows you the labels used in this column.

Using these codes, the stem *Umwohner* is given the code `PVx`, indicating that it is made up of a preposition, a verb, and an affix. The word *Abfahrtszeit* has the code `NxN`. The last five classes mentioned may cause some surprise since it may not be clear in which cases these labels are being used. A *c* indicating a contracted preposition is only used once in the database. The preposition *zur* in *zurzeit* is labeled as a *c*. Words like *Achtstundentag* can be analysed as `QNxN␣`which means that this word contains three stems in combination with an affix `(SSAS)`. These kind of Stem/Affix combinations are not part of the limited constructions which we consider to be legal. Therefore a new entity had to be introduced. This is a so-called *Node*. A node is a combination of two or more stems which as such can only be used in compounds with at least one other stem. *Achtstunde* does not mean anything unless it is used in combination with a word like *Tag* or *Woche*. The *p* is used for a Node-like construction in which the two parts, like *Aussenbord* in *Aussenbordmotor*, are formed by a preposition combined with a noun. Some other examples are *Nachhauseweg*, *Unterseeboot* and *Untertagearbeiter*. The last label *Root* is used in those cases in which two or more words are obviously related, but it is hard to tell from which word they derived. Obviously, *Demonstrant* and *Demonstration* have something in common. One might say that the verb *demonstrieren* can be seen as the basis for both words. However in some cases it is more difficult to tell which word should be considered to be the basic word. Therefore the part *demonstr* is called the *root*. Together with the suffix *ation* or *ant* the words *Demonstration* and *Demonstrant* can easily be analysed.

The FLEX name and description of the column that gives you these codes are as follows:

ImmClass (ImmClassLemma) `Immediate␣segmentation,␣word␣class␣labels␣`

The third immediate segmentation column simply tells you whether the elements identified are stems or affixes. Upper case `S␣`indicates a stem, upper case `A␣`indicates an affix. Thus the stem *Absichtserklaerung* is represented as `SASA`. The FLEX name and description of this column are as follows:

ImmSA (ImmSALemma) `Immediate␣segmentation,␣stem/affix␣labels␣`

The fourth immediate segmentation column concerns stem allomorphy. Within derived words or compounds, stems sometimes take a form different from their forms found in isolation. These changes may involve replacement of the stem vowel or the inclusion or deletion of one or more consonants. When morphological analysis is noted down, any resulting stems are given their normal stem form, because that is the most appropriate form which occurs in German. An example is the word *Abbruch*, which comprises the affix *ab* and the stem *brech*: note the difference between *bruch* and *brech*, where the one element is spelt two different ways. This is called stem allomorphy. If allomorphy takes the form of adding or dropping an Umlaut, this is indicated seperately in the column described below. This column indicates whether or not stem allomorphy occurs in its immediate segmentation. The code `Y␣`means that it does occur, the code `N␣`that it does not. The FLEX name and description for this column are as follows:

ImmAllo `Stem␣allomorphy,␣top␣level␣`
(ImmAlloLemma)

The fifth column identifies those words whose analysis is *opaque* – that is, words made up of morphemes which are recognizable, but where the meaning of the head element isn't reflected in the meaning of the full word. An example of this is *Angsthase*: it appears to be made up of the noun *Angst* and the noun *Hase* (the head element). Since the semantic link between *Hase* and *Angsthase* is far from obvious, the analysis is marked as being opaque, and it gets a `Y␣`in this column. Words whose analyses are morphologically and semantically clear get the code `N`. The FLEX name and description of this column are as follows:

ImmOpac `Opacity,␣top␣level␣`
(ImmOpacLemma)

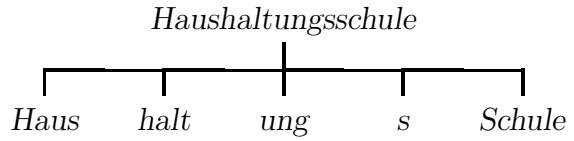
The last of the six immediate segmentation columns marks those stems whose morphological analysis involves *Umlaut*. This is the process whereby a vowel of one of the morphemes changes in the process of compounding or derivation. For example, *Anwältin* is analysed as the stem *Anwalt* and the affix *-in*: the stem has changed from *Anwalt* to *Anwält* when the female equivalent of the word *Anwalt* is constructed by

adding the suffix *in*. In this case the sixth column gives `Y␣`for yes if a vowel mutation of one of the vowels of the morphemes take place. The FLEX column name and description of this column are as follows:

ImmUml `Umlaut,␣top␣level␣`
(ImmUmlLemma)

3.5.2 COMPLETE SEGMENTATION (FLAT)

Complete segmentation is 'complete' in the sense that it identifies all the morphemes a stem contains. This is in contrast to immediate segmentation, which only picks out the next two (sometimes three or four) morphological elements. The complete segmentation discussed in this section is also *flat*, which means that you can see what the constituent morphemes are without knowing the details of the full morphological analysis which has been carried out. When you draw a morphological 'tree diagram', this information gives the outermost branches only; you cannot analyse any further, and you cannot see the intermediate levels. So, when you want to see the complete, flat, segmentation of *Haushaltungsschule* for example, you get this sort of information:



There are three columns with complete segmentation (flat) information. The first contains the morphemes themselves. The second contains the word class of each morpheme, and the third simply states whether each morpheme is a stem or an affix. The last two columns are useful when you're looking for a stem with a particular combination of morphemes: using the FLEX `SHOW␣`and `QUERY␣`options, you can hunt out stems which are made up of a noun plus an affix plus a noun, say, or all the stems which contain at least three other stems.

The first column gives you each stem split into its morphemes by `+␣`signs. Thus the stem *Haushaltungsschule* is written in the following way:

`Haus+halt+ung+s+Schule␣`

No diacritics are included. The FLEX name and description of this column are as follows:

Flat (FlatLemma) `Flat␣segmentation␣`

The second column uses single-letter codes to represent the word class of each morpheme. Using these codes, the stem *Haushaltungsschule* is given as `NVxxN`. The FLEX name and description of the column are as follows:

FlatClass (FlatClassLemma) `Flat␣segmentation,␣word␣class␣labels␣`

Word Class	Label
Adjective	`A`
Adverb	`B`
Conjunction	`C`
Article	`D`
Lexicalized Flection	`F`
Interjection	`I`
Noun	`N`
Pronoun	`O`
Preposition	`P`
Quantifier/Numeral	`Q`
Root	`R`
Verb	`V`
Affix	`x`

Table 8: Word class labels (flat segmentation)

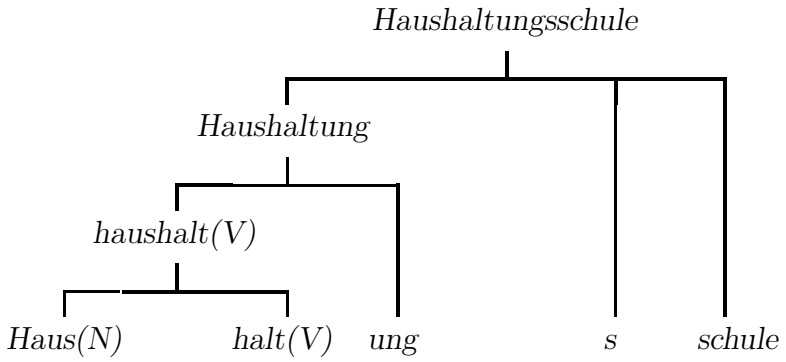
The last column simply indicates whether each morpheme is a stem or an affix. Upper case `S␣`means Stem, and upper case `A␣`means Affix. The full code for *Haushaltungsschule* is thus `SSAAS`. The FLEX name and description of this column are as follows:

FlatSA (FlatSALemma) `Flat␣segmentation,␣stem/affix␣labels␣`

3.5.3 COMPLETE SEGMENTATION (HIERARCHICAL)

Complete, hierarchical segmentation gives the most detailed analysis available for each stem. It is called *hierarchical*

because it can cover several different levels: it is arrived at after immediate analysis has been carried out on every stem that can be identified within a larger stem. With this information, you can draw a complete morphological 'tree diagram', from the root to the outermost branches, with every intermediate branch fully represented. So, for the stem *Haushaltungsschule*, you can get the following morphological analysis:



There are six columns which give information about the full segmentations of stems. Three of them give the hierarchical segmentations themselves. The simplest of these tells you what the constituent morphemes of the stem are, indicating with algebra-like brackets the structure of the 'tree'. Also available are similar bracket notations which supply a word class label alongside each morpheme on each level, or the word class without the morpheme itself. The remaining three columns indicate whether stem allomorphy, vowel mutation (Umlaut) or a change of meaning (Opacity) occurs in the full hierarchical analysis.

The first column provides all the information you need to draw a tree diagram like the one above – that is, the constituent morphemes of a stem each delimited by a comma and enclosed in brackets which indicate its complete morphological structure. The stem *Haushaltungsschule* thus looks like this:

`((((Haus),(halt)),(ung)),(s),(Schule))`␣

Each identifiable stem or affix is enclosed by a pair of brackets, beginning with the brackets round the full original stem. Then there is a pair of brackets round each of the two elements of the derivation *Haushaltung* one more pair around

the compound *Haushalt*, and finally a pair of brackets round each of the five constituent morphemes.

The FLEX name and description of the column which contains morphological analyses in this form are as follows:

Struc `Structured␣segmentation␣`
(StrucLemma)

The next two columns use extra labels to indicate the word class of each segment. They are given between square brackets to the right of each closing round bracket, so that every segment on every level within the original stem has a word class code. The word class codes used are as follows:

Word Class	Label
Noun	`N`
Adjective	`A`
Quantifier/Numeral	`Q`
Verb	`V`
Article	`D`
Pronoun	`O`
Adverb	`B`
Preposition	`P`
Conjunction	`C`
Interjection	`I`
Abbreviation	`X`
Lexicalized Flection	`F`
Root	`R`

Table 9: Word class labels (complete segmentation)

The codes used for affixes are combinations of these word class labels. The stem *Haushaltungsschule* can be represented as follows:

```
((((Haus)[N],(halt)[V])[V],(ung)[N|V.])[N],(s)[N|N.N], (Schule)[N])[N]
```

This example illustrates the special form affix codes take. There are two elements in each affix code which are separated by a vertical bar `|`. In front of the vertical bar is a single code which is the word class of the stem which the affix in question helps to form. After the vertical bar comes a combination of single letter codes which indicate the word class of each element within the stem formed, and the position of the affix itself is given by a dot.

In the *Haushaltungsschule* example above, the code given alongside the affix *ung* is `[N|V.]`. The `N␣`before the bar means that the affix *ung* helps to form a stem which is a noun *(Haushaltung)*. The `V.␣`after the bar means that the segmentation of the noun *Haushaltung* is verb plus affix. These detailed codes can help you to identify the way affixes are used, and to get lists of stems which contain affixes used in particular contexts: the fact that the second part of the *ung* code is `V.␣`helps you to see at once that this affix helps to form a derivation, in conjunction with a verb.

Sometimes a pair of affixes can only be used together, as in the word *Gebirge* – the word *birge* does not exist and the word *Gebirg* does not exist. In such cases, `x␣`marks the other part of the affix, and denotes that the affixes must occur in combination with each other: so-called *split affixes*. The code for the *ge-* of *Gebirge* is thus `[N|.Nx]`, and the code for the *-e* is `[N|xN.]`.

So, this column is particularly useful for two things. First, you can see the word class of each stem in the segmentation alongside the orthographic representations of individual morphemes. Second, you get detailed information about each affix each stem contains. The FLEX name and description of this column are as follows:

StrucLab `Structured␣segmentation,␣word␣class␣labels␣`
(StrucLabLemma)

The next column shows the hierarchical structure of each stem by means of round brackets and commas, and the full word class labels between square brackets, just as with the previous column. The only difference is that in this column the orthographic representation of the constituent stems and affixes is missed out altogether. Thus the stem *Haushaltungsschule* gets the following representation:

`((((()[N],()[V])[V],()[N|V.])[N],()[N|N.N],()[N])[N]`

This column again helps you to search for stems which have a particular morphological structure and particular combinations of syntactic elements. The FLEX name and description of this column are as follows:

StrucBrackLab `Structured␣segmentation,␣word␣class␣labels␣only␣`
(StrucBrackLabLemma)

The fourth hierarchical segmentation column deals with stem allomorphy. Within derived words or compounds, stems sometimes take a form different from their forms found in isolation. These changes may involve replacement of the stem vowel, or the inclusion or deletion of one or more consonants. When a morphological analysis is noted down, the resulting stems are given their normal stem orthography, because that is the most appropriate form which occurs in German. An example is the word *Abbruch*, which comprises the affix *ab* and the stem *brech*: note the difference between *bruch* and *brech*, where the one element is spelt two different ways. This is stem allomorphy. If allomorphy takes the form of adding or dropping an Umlaut, this is indicated separately in the column described below. This column indicates whether or not stem allomorphy occurs at any point in a stem's complete hierarchical segmentation. The code `Y␣`means that it does occur, the code `N␣`that it does not. The FLEX name and description for this column are as follows:

StrucAllo `Stem␣allomorphy,␣any␣level␣`
(StrucAlloLemma)

The fifth column identifies those words whose analysis is *opaque* – that is, words made up of morphemes which are recognizable, but where the meaning of the head element isn't reflected in the meaning of the full word. An example of this is *Angsthase*: it appears to be made up of the noun *Angst* and the noun *Hase* (the head element). Since the semantic link between *Hase* and *Angsthase* is far from obvious, the analysis is marked as being opaque, and it gets a `Y␣`in this column. Words whose analyses are morphologically and semantically clear get the code `N`. The FLEX name and description of this column are as follows:

StrucOpac `Opacity,␣any␣level␣`
(StrucOpacLemma)

The last of the six hierarchical segmentation columns marks those stems whose morphological analysis involves *Umlaut*. This is the process whereby a vowel of one of the morphemes changes in the process of compounding or derivation. For example, *Anwältin* is analysed as the stem *Anwalt* and the affix *-in*: the stem has changed from *Anwalt* to *Anwält* when the female equivalent of the word *Anwalt* is constructed by

adding the suffix. The FLEX column name and description of this column are as follows:

StrucUml `Umlaut,␣any␣level␣`
(StrucUmlLemma)

3.6 OTHER CODES

The remaining three columns give counts of various sorts: the number of *components* (i.e. stems and affixes) in the immediate analysis of each stem, the number of *morphemes* each stem contains, and the number of *levels* involved in the complete hierarchical analysis of each stem.

The first of these columns is the simple count of the number of components each stem contains. The normal figure is two; words are generally split into two parts each time one level of morphological analysis takes place. Sometimes three components can be identified: Derivational compounds are usually analysed as a stem plus a stem plus an affix, as are normal compounds which are joined with any 'link morpheme'. Derivational compounds occasionally contain four elements, stem plus affix plus stem plus affix. And of course, monomorphemic words only contain one component. Any stems which have not yet received an adequate morphological analysis (for the reasons given in section 3.1.3) get the number 0.

Some examples: the number of components in the stem *Abhängigkeitsverhältnis* is three *(Abhängigkeit + s + Verhältnis)*, and for the stem *Haustür* it is two *(Haus + Tür)*.

The FLEX name and description of this column are as follows:

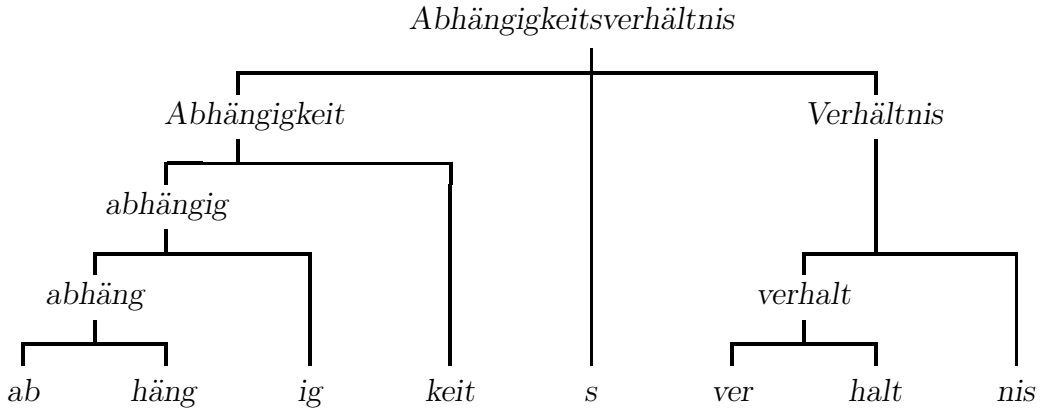
CompCnt `Number␣of␣morphological␣components␣`
(CompCntLemma)

The second column gives you the number of morphemes in each stem. For words without a morphological analysis, the number given is zero. The number of morphemes in the stem *Abhängigkeitsverhältnis* for example is eight, while for *Haustür* it is two.

The FLEX name and description of this column are as follows:

MorCnt `Number␣of␣morphemes␣`
(MorCntLemma)

The last of the three columns gives a count of the number of levels in the complete hierarchical segmentation described above, which is best illustrated by means of a tree diagram:



Including the stem at the top, the diagram covers five lines: this is the *number of levels* the stem has. It is the number of times you can carry on doing immediate analysis when you analyse a particular stem in full. Do not confuse it with the number of all the immediate analyses required to arrive at the complete hierarchical segmentation (which for *Abhängigkeitsverhältnis* is six); any one *level* of analysis may include more than one immediate segmentation. Monomorphemic stems always get the number 1, while stems without analysis (for reasons explained in section 3.1.3) get the number 0.

The FLEX column name and description of this column are as follows:

LevelCnt `Number␣of␣morphological␣levels␣`
(LevelCntLemma)

3.7 MORPHOLOGY OF GERMAN WORDFORMS

There are two types of morphology information available for the 360,000 wordforms given in the CELEX database: first, information about the lemma which underlies each family of wordforms, and second, a simple identification of the inflectional features which are specific to each wordform, either in the form of twenty-nine 'yes/no' feature columns or one column with feature identification codes.

Dictionaries present their lexical information under boldtype headwords, which are used instead of listing every individual inflected form separately. Such a form is often called

the *canonical form*, since it represents a full canon of inflections. Thus the word *esse* is understood as referring not only to the form *esse* itself, but also the forms *essen, gegessen, aß*, and *aßen* and a host of others. To print full details about every inflected form separately would result in a lot of needless repetition and enormous books which no one could lift from the bookshelf. However, for many applications, lemma information has to be listed for each individual wordform, and in a CELEX lexicon of type wordform, you can do just that when you include certain 'morphological' columns. This is done by providing a link between the wordform information and the lemma information. When you choose the option `Lemma␣information␣`from the `ADD␣COLUMNS␣`menu, you are in fact being allowed into the lemma information by the back door. You can now look up information specific to a particular wordform in your lexicon, and at the same time see general information which is common to all the other forms in the same inflectional paradigm. One particularly useful type of lemma information you can use in your wordform lexicon is the syntactic information, which can give the word class of any wordform you are looking at. There is also an important distinction which you may be able to draw upon with the frequency information. The wordform lexicon gives you a MANNHEIM frequency figure specific to each wordform, while the lemma information available lets you see the sum frequency for all the inflectional forms in the same paradigm, a figure referred to as the *lemma frequency*.

All the lemma information has already been defined elsewhere in this linguistic guide, so there is no point in repeating it all here. All that needs to be pointed out is that the column names used in a real lemma lexicon differ from those used in the lemma information option in the morphology of wordforms. When a FLEX column name and description are defined in the course of lemma lexicon text, the column name given in brackets is the name of the column when it is used as part of a wordforms lexicon. Usually this name is identical to the lemma lexicon name, except that the word *lemma* is added to the end.

ExampleName (ExampleNameLemma)

```
The␣column␣names␣used␣for␣lemma␣information␣
in␣a␣Wordforms␣lexicon␣are␣given␣in␣
brackets,␣as␣this␣Example␣Name␣shows.␣
```

All the other details and definitions remain the same in both cases. So, when you're looking for the columns of lemma information provided with a wordforms lexicon under morphology, just go back to the original lemma information: it's all there.

3.7.1 INFLECTIONAL FEATURES

There are twenty-nine special columns available only with a lexicon of type wordforms. Each one corresponds to a particular inflectional attribute which a wordform can have. There can only be one of two codes in each column: `Y␣`for 'yes, this wordform has this attribute', or `N␣`for 'no, this wordform does not have this attribute'. These columns are therefore useful for constructing restrictions on your lexicons, restrictions which need not be 'on view': it's unlikely that you will want to look at the contents of these columns with the `SHOW␣`option. (If, on the other hand, you want to have a label which lets you see at a glance all the inflectional features each wordform has, then you should use the 'type of flection' codes described in the next section.)

An example. To make a lexicon which gives you all the wordforms in the database with the exception of the 'separated' forms of verbs, you have to include at least two columns in the wordforms lexicon you create, namely a column which gives the orthographic representations you prefer, and ***Sepa*** (which is amongst the twenty-nine columns described below). You must then construct a restriction for your lexicon which states that ***Sepa*** must be equal to `N`. You can then format your lexicon to make sure that ***Sepa*** is not 'on view': that way, when you `SHOW␣`or `EXPORT␣`your lexicon, you just get the list of words you require without the list of `N`'s. To this basic lexicon, you can of course add any other columns you require, either the orthographic and frequency information specific to each wordform, or the general lemma information—particularly syntax—which is available through the 'Morphology of German wordforms' options.

The first inflectional features column marks those wordforms which have two separate parts, even though they 'belong' to a stem or headword which is a single unit. Forms like *achtete hoch*, *ackert durch* and *addiert auf* have the positive `Y␣`code,

even though their headwords are *hochachten*, *durchackern*, and *aufaddieren*. The FLEX name and description of this column are as follows:

Sepa `Separated␣wordform␣`

The second column indicates whether a wordform is a singular form of any sort. Mostly this means verbal forms such as *lauf* or *höre auf*, or nouns such as *Fahrrad*. The FLEX name and description of this column are as follows:

Sing `Inflectional␣feature:␣singular␣`

The third column indicates whether a wordform is a plural inflection of any sort. Mostly this means verbal forms such as *laufen* or *hören auf*, or nouns such as *Fahrräder*. The FLEX column name and description of this column are as follows:

Plu `Inflectional␣feature:␣plural␣`

The fourth column indicates whether a wordform is a nominative inflection of a noun. Together with the information presented in the third column you are able to see whether this word is a word in its nominative singular or nominative plural form. Not only nouns are marked with a 'Y' if the wordform presented is a word in its nominative form but also pronouns like *ich* or *wer* and articles like *der* and *die*. The FLEX column name and description of this column are as follows:

Nom `Inflectional␣feature:␣nominative␣`

The fifth column indicates whether a wordform is a genitive inflection of a noun. Together with the information presented in the third column you are able to see whether this word is a word in its genitive singular or genitive plural form. Not only nouns are marked with a Y if the wordform presented is a word in its genitive form but also pronouns like *meiner* or *wessen* and articles like *des* and *der*. The FLEX column name and description of this column are as follows:

Gen `Inflectional␣feature:␣genitive␣`

The sixth column indicates whether a wordform is a dative inflection of a noun. Together with the information presented in the third column you are able to see whether this word is a word in its dative singular or dative plural form. Not only nouns are marked with a Y if the wordform presented is a word in its dative form but also pronouns like *mir* or *wem* and articles like *dem* and *der*. The FLEX column name and description of this column are as follows:

Dat `Inflectional␣feature:␣dative␣`

The seventh column indicates whether a wordform is an accusative inflection of a noun. Together with the information presented in the third column you are able to see whether this word is a word in its accusative singular or accusative plural form. Not only nouns are marked with a Y if the wordform presented is a word in its accusativ form but also pronouns like *mich* or *wen* and articles like *den* and *die*. The FLEX column name and description of this column are as follows:

Acc `Inflectional␣feature:␣accusative␣`

The eighth column marks all the wordforms which are positive forms – that is, not comparative or superlative forms like *besser* and *beste*, but plain adjectival forms like the word *gut*. Thus adjectives like *hoch* and *hohe* or *dumm* and *dumme* get the code `Y`, while all other forms get the code `N`. The FLEX name and description of this column are as follows:

Pos `Inflectional␣feature:␣positive␣`

The ninth column marks all the wordforms which are comparative forms. Adjectival wordforms such as *besser* or *erfolgreichere* thus get the code `Y`, while all other non-comparative forms get the code `N`. Possible adverbial comparative forms are listed as separate lemmas without any 'Y' values in this column. The FLEX name and description of this column are as follows:

Comp `Inflectional␣feature:␣comparative␣`

The tenth column marks all adjectival superlative forms, so that wordforms such as *best* or *größt* get the code `Y`, and every other form gets the code `N`. Possible adverbial superlative forms are listed as separate lemmas without any 'Y' values in this column. The FLEX column name and description of this column are as follows:

Sup `Inflectional␣feature:␣superlative␣`

The eleventh column marks the form of the verb usually known as the infinitive. It is used as a headword in the CELEX databases, and in most dictionaries. For most verbs, the ending is *-en*: *haben* or *fahren*, for example. Some other verbs have slightly different infinitives, such as *sein* or *tun* and *klettern*. Any wordform which is an infinitive gets a `Y␣` code in this column; all the others get the code `N`. The FLEX column name and description for this column are as follows:

Inf `Inflectional␣feature:␣infinitive␣`

The twelfth column marks all those wordforms which form the infinitive of a verb with an additional preposition *zu*. This always occurs in the case of separable verbs. For example: *abzuarbeiten* and *abzubauen* get a `Y␣`code in this column; all the others get the code `N`. The FLEX column name and description for this column are as follows:

ZuInf `Inflectional␣feature:␣infinitive␣with␣"zu"␣`

The thirteenth column marks any participles, past tense or present tense. Present participles are normally formed by adding *-(e)nd* to the stem of the verb, with the exception of some irregular verbs. Past participles of 'weak' verbs add the prefix *ge-* and the suffix *-(e)t* to the stem, and they are used in the formation of the perfect tense: 'Ich habe zwei Jahre in Berlin *gearbeitet*'. The past participle of a 'strong' verb, conversely, ends in *-en*, while a vowel change may also occur within the stem itself: 'ich habe zu viel *getrunken*'. Most past participles can also be used adjectivally, as in 'das *gefaltete* Blatt'. Any wordforms which are participles get the code `Y`, and all the rest get the code `N`. The FLEX name and description of this column are as follows:

Part `Inflectional␣feature:␣participle␣`

The fourteenth column identifies any present tense forms, including the present participles mentioned under ***Part***. Thus verb forms like *abbezahle, abbezahlen* and *abbezahlend* get the code `Y`, while all other forms (including infinitives, which are marked in a different column) get the code `N`. The FLEX name and description of this column are as follows:

Pres `Inflectional␣feature:␣present␣tense␣`

The fifteenth column identifies any past tense forms, including the past participles mentioned under ***Part***. In the simple past tense, regular 'weak' verbs add *-(e)tet*, *-(e)test*, *-(e)te* and *-(e)ten* to the stem, as in 'ihr *arbeitetet*' or 'du *hörtest*', 'er *arbeitete*' , 'wir *hörten*'. There are many other 'strong' verbs, which often just change a vowel sound in the stem, as in 'ich *schrieb* ein Buch'. All past tense forms get the code `Y`, while all other forms (including infinitives, which are marked in a different column) get the code `N`. The FLEX name and description of this column are as follows:

Past `Inflectional␣feature:␣past␣tense␣`

The sixteenth column marks first person singular forms of verbs, present and past, indicative and subjunctive. For most verbs, the present first person form is derived from the stem of the verb by adding an 'e', like in *ich gebe*. So, all first person singular forms, like 'ich *fahre*' or '*schlug nach*', are given the code `Y`. The FLEX column name and description of this column are as follows:

Sin1 `Inflectional␣feature:␣1st␣person␣verb␣`

The seventeenth column marks second person singular forms of verbs, present and past, indicative and subjunctive. For most verbs, the present second person form consists of the stem plus the suffix *-(e)st*. Also for some verbs there is a change in the stem vowel from *e* to *i* or *ie* or Umlaut mutation like the second person singular of the verb *geben* which is *gibst* or the second person singular of the verb *stehlen* which is *stiehlst*. So, all second person forms like 'du *schläfst*' or '*liefst* du?' are given the code `Y`. The FLEX column name and description of this column are as follows:

Sin2 `Inflectional␣feature:␣2nd␣person␣verb␣`

The eighteenth column identifies third person singular forms of the verb,present and past, indicative and subjunctive. For most verbs, the third person form consists of the stem plus the suffix *-(e)t*. Also for some verbs there is a change in the stem vowel from *e* to *i* or *ie* or Umlaut mutation like in the third person singular of the verb *geben* which is *gibt* or the third person singular of the verb *stehlen* which is *stiehlt*. Thus forms like 'Er *bleibt* dort' or 'Gilbert *schrieb*' or 'Er sagt, er *hoffe*, daß alles gut geht' get the code `Y`. The FLEX name and description for this column are as follows:

Sin3 `Inflectional␣feature:␣3rd␣person␣verb␣`

The nineteenth column identifies first and third person plural forms of the verb, again for both present and past tense, and indicative and subjunctive moods. Thus forms like 'Wir *lesen* viel' or 'Die Leute *standen* im strömenden Regen vor der geschlossenen Bahnhofshalle und warteten auf den Schnellzug nach Lodz, der für Sie die einzige Hoffnung war sich aus dieser miserablen Lage zu retten' get the code `Y`. The FLEX name and description for this column are as follows:

Plu13 `Inflectional␣feature:␣1st/3rd␣person␣plural␣verb␣`

The twentieth column identifies present and past, indicative and subjunctive. second person plural forms of the verb. Thus forms like 'Ihr *lest* viel' or 'Ihr *fandet* es doch nicht schlimm?' get the code `Y`. The FLEX name and description for this column are as follows:

Plu2 `Inflectional␣feature:␣2nd␣person␣plural␣verb␣`

The twenty-first column marks the indicative forms. Together with the columns ***Present Tense*** or ***Past tense*** it is possible to derive information about the so called *Indikativ Präsens* and the *Indikativ Präteritum*. An example of an *Indikativ Präsens* is 'ich *hoffe*, daß du kommst' and an *Indikativ Präteritum* 'Ich *fand* es nicht einfach.' These forms have the code `Y␣`in this column, while every other wordform gets the code `N`. The FLEX name and description of this column are as follows:

Ind `Inflectional␣feature:␣indicative␣`

The twenty-second column marks the subjunctive forms. Together with the columns ***Present Tense*** or ***Past tense*** it is possible to derive information about the so called *Konjunktiv Präsens* and the *Konjunktiv Präteritum*. An example of a *Konjunktiv Präsens* is 'man *nehme* täglich einen Liter Wein' and as *Konjunktiv Präteritum* 'Ich *hätte* dich bestimmt nicht geglaubt.' These forms have the code `Y␣`in this column, while every other wordform gets the code `N`. The FLEX name and description of this column are as follows:

Sub `Inflectional␣feature:␣subjunctive␣`

The twenty-third column marks the imperative form of a verb. An example of an imperative form is the word *Sei* in the sentence: '*Sei* doch mal still'. These wordforms that get the code `Y␣`in this column ; every other wordform gets the code `N`. The FLEX name and description for this column are as follows:

Imp `Inflectional␣feature:␣imperative␣`

The twenty-fourth column marks all (nominalized) adjectives, numerals or pronouns which have an inflectional *-e* ending like the words *wissenschaftliche* and *kalte*. So if a wordform ends in the inflectional *-e*, then it gets the code `Y␣` in this column, and all the other wordforms get the code `N`. The FLEX name and description of this column are as follows:

Suff_e `Inflectional␣feature:␣with␣suffix␣-e␣`

The twenty-fifth column marks all those (nominalized) adjectives, numerals or pronouns which have an inflectional *-en* ending like the words *großen* and *kleinen*. So if a wordform ends in the inflectional *-en*, then it gets the code `Y␣`in this column, and all the other wordforms get the code `N`. The FLEX name and description of this column are as follows:

Suff_en `Inflectional␣feature:␣with␣suffix␣-en␣`

The twenty-sixth column marks all those (nominalized) adjectives, numerals or pronouns which have an inflectional *-er* ending like the words *sicherer* and *aufwendiger*. So if a wordform ends in the inflectional *-er*, then it gets the code `Y␣`in this column, and all the other wordforms get the code `N`. The FLEX name and description of this column are as follows:

Suff_er `Inflectional␣feature:␣with␣suffix␣-er␣`

The twenty-seventh column marks all those (nominalized) adjectives, numerals or pronouns which have an inflectional *-em* ending like the words *abbruchreifem* and *trostlosem*. So if a wordform ends in the inflectional *-em*, then it gets the code `Y␣`in this column, and all the other wordforms get the code `N`. The FLEX name and description of this column are as follows:

Suff_em `Inflectional␣feature:␣with␣suffix␣-em␣`

The twenty-eighth column marks all those (nominalized) adjectives, numerals or pronouns which have an inflectional *-es* ending like the words *himmelhohes* and *freudiges*. So if a wordform ends in the inflectional *-es*, then it gets the code `Y␣` in this column, and all the other wordforms get the code `N`. The FLEX name and description of this column are as follows:

Suff_es `Inflectional␣feature:␣with␣suffix␣-es␣`

The twenty-ninth column marks all those (nominalized) adjectives, numerals or pronouns which have an inflectional *-s* ending like the words *eins* and *deins*. So if a wordform ends in the inflectional *-s*, then it gets the code `Y␣`in this column, and all the other wordforms get the code `N`. The FLEX name and description of this column are as follows:

Suff_s `Inflectional␣feature:␣with␣suffix␣-s␣`

3.7.2 TYPE OF FLECTION

In the 'Inflectional Features' section above, twenty-nine different inflectional features are distinguished, and assigned to twenty-nine separate 'yes/no' columns. The same information is also available in one single column, using combinations of single-letter codes to show all the features each wordform has. The 'yes/no' columns are useful for constructing restrictions on your lexicon, whereas the 'type of flection' column described here provides you with a label that identifies at a glance all the features each wordform has. Table 10 below sets out the single-letter codes.

For a full definition of these flection types, read the details given for the appropriate 'yes/no' columns in section above. However, note that there are type of flection labels which do not correspond to a 'yes/no' column. The `X␣`label identifies many forms not covered by the other labels, including adverbs like *damals*, prepositions like *seit* or conjunctions like *damit*. These forms are always the same as those used as the headword form of the lemma. No nouns, verbs or adjectives ever get the code `X`. The following three codes `m`, `w␣`and `s␣` are used to indicate the gender of a noun, pronoun or article. The last code is `0␣`which is the code for the uninflected form of an adjectival noun, numeral or pronoun, which is the base form of these categories.

Each wordform may have more than one code attached to it. Thus the wordform *Abbaurecht* has the code `nS,dS,aS`: `S␣`means it is a singular, `n␣`means that it is a nominative, `d␣` means that it is a dative and `a␣` means that it is an accusative. Similarly, the verbal wordform *hacken* is assigned the code '13PIE, 13PKE, i'. In other words, whenever more than one type of flection applies to a single orthographical form, distinct types are separated by commas.

The FLEX name and description of this column are as follows:

FlectType `Type␣of␣flection␣`

Inflectional feature	Label	'yes/no' column name
Separated wordform	/	***Sepa***
Singular	S	***Sing***
Plural	P	***Plu***
Nominative	n	***Nom***
Genitive	g	***Gen***
Dative	d	***Dat***
Accusative	a	***Acc***
Positive	o	***Pos***
Comparative	c	***Comp***
Superlative	u	***Sup***
Infinitive	i	***Inf***
Infinitive with 'zu'	z	***ZuInf***
Participle	p	***Part***
Present tense	E	***Pres***
Past tense	A	***Past***
1st person verb	1	***Sin1***
2nd person verb	2	***Sin2***
3rd person verb	3	***Sin3***
Indicative	I	***Ind***
Subjunctive only	K	***Sub***
Imperative	r	***Imp***
With suffix -e	4	***Suff_e***
With suffix -en	5	***Suff_en***
With suffix -er	6	***Suff_er***
With suffix -em	7	***Suff_em***
With suffix -es	8	***Suff_es***
With suffix -s	9	***Suff_s***
Headword form (not nouns, verbs or adjectives)	X	
masculine	m	
feminine	w	
neuter	s	
uninflected form		
adjectival declination	0	

Table 10: Type of flection labels

4 GERMAN SYNTAX

Syntactic information is available for lemma lexicons. It consists of syntactic codes which describe all the lemmas in the database. A general word class code is available, as well as more detailed codes on nouns, verbs, adjectives, numerals, pronouns and prepositions. Diagram 'Syntax of German Lemmas' in Appendix 1 gives an overview of the syntactic information offered to you in the ADD␣COLUMNS␣menus:

```
                ADD COLUMNS

Word class                              >
Subclassification nouns                 >
Subclassification verbs                 >
Subclassification adjectives            >
Subclassification numerals              >
Subclassification pronouns              >
Subclassification prepositions          >


TOP MENU
PREVIOUS MENU
```

If you want to use syntactic information of this type in conjunction with a wordforms lexicon (perhaps you want to know the word class of your wordforms), then you should use the 'lemma information' columns available with the morphological columns for wordforms. Since the syntactic category of a wordform is always the same as the lemma it belongs to, there is no need to provide extra, unnecessary syntactic columns for wordforms. The special link with lemma information means you can get access to all sorts of general information about the lemmas which represent each wordform.

However on occasions there are wordforms whose categorizations are different from those given for their lemma. Although the infinitive form of a verb can be used as a noun ('das *Schmeißen* von Zwergen ist nicht länger erlaubt') it is always classified as a verb. Such differences are specific to

certain wordforms, and because they usually work according to well-known rules, the details need not be given in the database.

4.0.1 SYNTACTIC CODES: LETTERS OR NUMBERS

For most of the classifications described below, there are two ways of representing each syntactic code. You can choose whether to use numbers (`Numeric␣codes`) or shortened verbal codes (`Labels`). An adverb, for example, is represented by the number `7␣`or the letters `ADV`. No matter which type of codes you decide to use, the *information* remains the same; only the *representation* changes.

Numeric codes use single digits to represent syntactic subclassifications. If ever you see a lemma with more than one digit, it means that more than one of the syntactic categories can apply to it. Thus the verb *abkühlen* for example, has the subclassification code 536: the 5 means 'this can be a lexical verb', and the 3 means 'this can be an impersonal verb' and the 6 means 'this can be a reflexive verb'. A null value (that is, no value at all) means that the particular subcategorization is not appropriate for the lemma in question.

Subcategory labels are made up of letters or short abbreviations. When a lemma fits more than one subcategory, the appropriate labels are simply linked up. Thus the verb *abkühlen* is given the subclassification label `lir`. This means that the lemma can be a lexical verb, an impersonal verb or a reflexive verb. A null value means that the particular subcategorization is not appropriate for the lemma in question.

4.1 WORD CLASS

The word class code is a simple way to identify the syntactic class of every lemma in the database. Ten basic categories – set out in Table 11 below – are distinguished, and you can identify them using either of the two forms described in section 4.0.1 above. Note that there are no null values in these columns: one of the categories listed is applied to every lemma.

The definitions of the two word class columns are given below, followed by Table 11 which sets out the meaning of each code with examples. If you want syntactic codes in the form

of numbers, choose the column with this FLEX name and description:

ClassNum (ClassNumLemma) `Word␣class,␣numeric␣`

If you want syntactic codes in the form of short verbal symbols, choose the column with this FLEX name and description:

Class (ClassLemma) `Word␣class,␣labels␣`

Word Class	Columns		Example
	ClassNum	***Class***	
Noun	1	N	*Haus*
Adjective	2	A	*klein*
Quantifier/Numeral	3	NUM	*mehr, sechs*
Verb	4	V	*abkühlen*
Article	5	ART	*das*
Pronoun	6	PRON	*ich*
Adverb	7	ADV	*anstandshalber*
Preposition	8	PREP	*von*
Conjunction	9	C	*und*
Interjection	10	I	*ach*

Table 11: Word class codes

One important distinction between nouns in German is *gender*. Using the information described here, you can find out the gender of any noun. In addition, *proper nouns* (names of various sorts) are further subclassified.

4.1.1 NOUNS: GENDER

There are three genders in German: *masculine*, *feminine*, and *neuter*. In addition to these three, CELEX also identifies those nouns which can be treated as masculine as well as feminine or neuter. This makes ten basic 'genders', which are represented by a set of numeric codes and a set of labels (as described in section 4.0.1 above). Table 12 below gives the meanings represented by both sets of codes along with some examples:

Gender	Columns		Example
	GendNum	***Gend***	
masculine	1	M	*Mann*
feminine	2	F	*Frau*
neuter	3	N	*Kind*
masculine/feminine	12	MF	*Sellerie*
masculine/neuter	13	MN	*Begehr*
feminine/masculine	21	FM	*Abgesandte*
fem./masc./neuter	213	FMN	*Dingsbums*
feminine/neuter	23	FN	*Beschwer*
neuter/masculine	31	NM	*Binokel*
neuter/feminine	32	NF	*Elastik*

Table 12: Nouns: gender codes

The FLEX names and descriptions of these ten gender code columns are as follows:

GendNum ***(GendNumLemma)*** `For␣nouns:␣gender,␣numeric␣`

Gend ***(GendLemma)*** `For␣nouns:␣gender,␣labels␣`

4.1.2 PROPER NOUNS

A proper noun is a name of some kind. CELEX distinguishes three types of proper nouns, and Table 13 defines these four types and gives examples:

Proper Noun	Columns		Example
	PropNum	***Num***	
Geographical names	1	G	*Amerika*
Names of people	2	P	*Amor*
Company or product names	3	B	*Baedeker*

Table 13: Proper noun codes

The two columns available with information on proper nouns contain codes in numeric forms or as labels (as described in section 4.0.1), and their FLEX names and descriptions are as follows:

PropNum (PropNumLemma) `For␣nouns:␣proper␣noun,␣numeric␣`

Prop (PropLemma) `For␣nouns:␣proper␣noun,␣labels␣`

4.1.3 SINGULARIA TANTUM

In German there are, as well as in other languages, nouns of which only the singular form exists. Words like *Hagel* or *Schnee* are examples of singularia tantum. For those nouns this column includes the code `Y`. The FLEX name and description are as follows:

SingTant (SingTantLemma) `For␣nouns:␣singulare␣tantum␣`

4.1.4 PLURALIA TANTUM

In German there are, as well as in other languages, nouns of which only the plural form exists. Words like *Ferien* or *Geschwister* are examples of pluralia tantum. For those nouns this column includes the code `Y`. The FLEX name and description are as follows:

PlurTant (PlurTantLemma) `For␣nouns:␣plurale␣tantum␣`

4.2 SUBCLASSIFICATION VERBS

When the simple word class code isn't detailed enough, further information on verbs is available here. You can find out which verbs take *haben* as their auxiliary verb, which take *sein*, and which can take either *haben* or *sein*. In addition, different types of verbs are distinguished and coded – copulas, impersonal verbs, and ordinary lexical verbs, for example. Furthermore, detailed complementation codes are given for

each verb. As with all the syntactic information, both numeric codes and verbal labels (see section 4.0.1) are provided for each subclassification, except for verb complementation, which is represented by means of alphanumeric strings only.

4.2.1 PERFECT TENSE (HABEN/SEIN)

When the perfect tense occurs in German, one of two *auxiliary* verbs is linked with a main verb. (In the sentence *ich habe geschlafen*, for example, the main verb *schlafen* is supported by the auxiliary verb *haben*.) To find out whether the verb you have selected takes *haben* or *sein* in the perfect tense, include in your lexicon one of the columns described here. Table 14 below sets out the simple codes used in the two columns available. When either *haben* or *sein* can be used in conjunction with a particular verb, the codes for each auxiliary are combined to make a two-digit code.

Auxiliary	Columns		Example
	AuxNum	***Aux***	
haben	1	haben	*tun*
sein	2	sein	*wachsen*
haben or sein	12	haben/sein	*abbiegen*

Table 14: Perfect tense auxiliary verb codes

The FLEX names and descriptions of these two columns are as follows:

AuxNum ***(AuxNumLemma)***
`For␣verbs,␣auxiliary␣verb,␣numeric␣`

Aux ***(AuxLemma)***
`For␣verbs,␣auxiliary␣verb,␣labels␣`

4.2.2 SUBCLASSES

To distinguish further between all the verbs in the database, six subclassification codes are given in the two columns described here. The first category, *auxiliary verb*, is used in a sentence to modify the meaning of the lexical verb by adding distinctions in tense, aspect or voice. The second

category, *copula*, is also a function word, although it can occur independently in the verb phrase: it usually links a subject to a complement. An example is the sentence 'Bist du der Schuldige?', where the copula verb *sein* links the subject *du* to a complement *der Schuldige*. The third category, *impersonal verbs*, refers to those verbs which cannot have a referential subject; *es regnet*, for example. The fourth category, *modal verbs*, refers to those verbs which modify the meaning of the lexical verb by adding distinctions in mood, such as possibility, obligation or permission. In German there are six verbs that can be *modal verbs* if they appear in a sentence in combination with an infinitive. The fifth category, *lexical verb*, is a normal 'content word' verb; it is used in a sentence primarily for the meaning it conveys, rather than fulfilling a purely grammatical or structural role. The sixth category *reflexive verb* are verbs that can or must be used along with a reflexive pronoun, so that the pronoun and the subject of a sentence refer to the same entity, e.g. 'manchmal fühle ich mich überhaupt nicht wohl'

Often, a particular verb may get more than one code: the verb *regnen* is classified as an ordinary lexical verb *and* an impersonal verb, and thus has the numeric code `53␣`and the label '`li`'. Other verbs may require a different combination of the six basic codes.

The next table sets out the basic codes used, and after that, the FLEX names and descriptions for the two columns are given.

Subclass	Columns		Example
	SubClassVNum	***SubClassV***	
Auxiliary verb	1	a	*haben*
Copula	2	c	*bleiben*
Impersonal verb	3	i	*regnen*
Modal verb	4	m	*dürfen*
Lexical verb	5	l	*abwaschen*
Reflexive verb	6	r	*beherrschen*

Table 15: Verb subclass codes

The FLEX names and descriptions of these two columns are as follows:

SubClassVNum `For␣verbs,␣subclasses,␣numeric␣`
(SubClassVNumLemma)

SubClassV `For␣verbs,␣subclasses,␣labels␣`
(SubClassVLemma)

4.3 VERB COMPLEMENTATION CODES

In the FLEX item `Subcategorization␣lexical␣verbs␣`nine forms of possible verb complements are discussed. For all verbs in these nine columns all the possible verbal complements are indicated. Instead of giving Yes/No values as marks for complements of a verb there are four possible codes:

Code	Meaning
I	impossible
O	obligatory
P	possible
U	undetermined

Table 16: Verb complementation codes

So if for example a verb like *abklopfen* is selected, then the columns for accusative compliment, dative complement and prepositional complement state that all three of them are possible (code `P`), whereas the other verbal complements are impossible (code `I`). The column *Complete complementation* is used as an additional column which gives the information of the nine columns in an alternative representation.

4.3.1 COMPLETE COMPLEMENTATION

In order to be able to see the possible combinations of the nine columns to be discussed in the following subsections, the column *Complete complementation* contains a code that represents the complementation pattern of the verb. Every code of a particular verb is a frame containing 9 slots, each indicating whether the complement mentioned at that position is obligatory (indicated by a capital), optional (indicated by a lowercase letter) or unrealised (indicated by a zero).

Each slot in the frame corresponds to the realisation of a particular complement function:

Position	Meaning
1	Subject, always empty unless it is "es"
2	Subject complement
3	Accusative complement
4	Second accusative complement
5	Dative complement
6	Genitive complement
7	Prepositional complement
8	Second prepositional complement
9	Adverbial complement

Table 17: Positions for functions of complements

If for any reason the information is not available for this verb the code will be a string of nine question marks. If there is no complement at all then the string will contain nine zeros. On these nine positions seven codes can appear indicating the kind of realisation of this complement. The following codes are used:

Code	Meaning
N/n	Noun phrase
E	Empty subject "es"
A/a	Adverb phrase or prepositional phrase
G/g	Noun phrase or adjective phrase
Z/z	Zu-infinitive
I/i	Infinitive (bare)
P/p	Prepositional phrase

Table 18: Realisation of complements

In this table capitals are used to indicate that a complement is obligatory and lowercase letters are used if the complement is optional. It seems as if there are two codes for noun phrases, i.e. 'N' and 'G'. We chose to include the code G (derived from the German term "Gleichsetzungsnominativ"), which is the code for copular verbs requiring an additional noun phrase or adjective phrase in the nominative case. An example of such a verb is 'sein'. In the sentence *er is der Vater*, the noun phrase *der Vater* is an example of a "Gleichsetzungsnominativ", whereas it is also possible to build a sentence like *er ist schuldig*. In this case the complement of the verb is an adjective phrase.

Although the ninth slot of the frame is normally either zero or

filled with an uppercase or lowercase 'A' there are also eight more detailed labels which are used to bring out the semantic functions of the adverbial complement, if this appeared to be typically associated with the verb.

Code	Meaning
L	Locative adverb or prep phrase
T	Temporal adverb or prep phrase
M	Manner adverb or prep phrase
C	Causative adverb or prep phrase
U	Purpose adverb or prep phrase
S	Instrumental adverb or prep phrase
O	Comitative preposition phrase
R	Role preposition phrase

Table 19: Realisation for adverbials

Apart from adverb phrases, these codes are also used for prepositional phrases. So for every adverb in the table there is an alternative prepositional phrase.

All possible combinations in the code for complete complementation, with an illustartive example can be found in appendix Table of conjugations of German Verbs. Here we will take the verb *abklopfen* as an example. The complete complementation of this verb is presented by FLEX with the code: 00N0n0000; 00N000000; 00N000P00; 00n000000; 000000000;

The first code **00N0n0000** indicates that the verb *abklopfen* can be used in a sentence with an obligatory accusative complement and an optional dative complement. Such as in the sentence: "Ich klopfe dem Mann den Staub ab."

The second code **00N000000** states that the verb can also be used in a sentence with just an obligatory accusative complement. Such as in the sentence: "Ich klopfe den Mantel ab."

We realise that it would have been possible to derive the fact that the dative complement can be omitted because the first code already says so. Therefore code 2 can be ignored as well as code 5 **000000000** can be ignored because of the fact that code 4 **00n000000** already states that the accusative complement is an optional complement. Both have however been included, because they are associated with meaning variants reflected by different subentries in

dictionaries. Thus in the first complementation frame of *abklopfen* the entity denoted by the accusative object is itself removed, while in the second frame something is removed from the thing denoted by the accusative object.

The third code allows sentences like "Wir werden die Argumente auf ihre Stichhaltigkeit hin abklopfen", in which there is a prepositional complement as well as an accusative complement.

The FLEX name and description of this column is as follows:

CompComp (CompCompLemma) `For␣verbs,␣complete␣complementation␣`

In the following nine subsections the individual complements will be discussed briefly.

4.3.2 EMPTY SUBJECT

The first digit in the string of digits is only filled when the sentence contains an empty subject. In German the word 'es' is used to build a sentence with an empty subject, as in: "Es regnet jetzt schon vier Stunden." Although all other verbs take a fully referential subject, this will not be shown in the string of digits.

The FLEX name and description of this column is as follows:

CompEsSubj (CompEsSubjLemma) `For␣verbs,␣Es␣Subject␣`

4.3.3 SUBJECT COMPLEMENT

The second digit in the string of digits is filled in those sentences in which a copula is followed by a complement providing additional information about the subject. A verb with an additional subject complement is the verb *sein* which can, as well as other copulas, be the main verb of a sentence like: "Frank is der Täter." The fact that *Täter* appears in this sentence in its nominative case form already indicates that this is a co-referential with the subject.

The FLEX name and description of this column is as follows:

CompSubj (CompSubjLemma) `For␣verbs,␣subject␣complement␣`

4.3.4 ACCUSATIVE OBJECT

The third digit in the string of digits is filled in those sentences in which an accusative object is triggered by the verb. A verb with an accusative object is the verb *sehen*. In a sentence like "ich sehe das Mädchen." the noun phrase *das Mädchen* is an instance of the accusative object triggered by the verb *sehen*.

The FLEX name and description of this column is as follows:

CompAcc `For␣verbs,␣accusative␣object␣`
(CompAccLemma)

4.3.5 SECOND ACCUSATIVE OBJECT

The fourth digit in the string of digits is filled in those sentences in which next to the first accusative object there is a second accusative object triggered by the verb. A verb with a second accusative object is the verb *lehren*. In a sentence like "ich lehre das Kind die niederländische Sprache", the noun phrase *die niederländische Sprache* is an instance of a second accusative object triggered by the verb *lehren*.

The FLEX name and description of this column is as follows:

CompSecAcc `For␣verbs,␣second␣accusative␣object␣`
(CompSecAccLemma)

4.3.6 DATIVE OBJECT

The fifth digit in the string of digits is filled in those sentences in which a dative object is triggered by the verb. A verb with a dative object is the verb *geben*. In a sentence like "ich gebe dem Mädchen den Ball", the noun phrase *dem Mädchen* is an instance of the dative object triggered by the verb *geben*.

The FLEX name and description of this column is as follows:

CompDat `For␣verbs,␣dative␣object␣`
(CompDatLemma)

4.3.7 GENITIVE OBJECT

The sixth digit in the string of digits is filled in those sentences in which a genitive object is triggered by the verb. A verb with a genitive object is the verb *sein*. In a sentence like "er ist arabischer Abstammung", the noun phrase *arabischer Abstammung* is an instance of the genitive object triggered by the verb *sein*.

The FLEX name and description of this column is as follows:

CompGen (CompGenLemma) `For␣verbs,␣genitive␣object␣`

4.3.8 PREPOSITIONAL OBJECT

The seventh digit in the string of digits is filled in those sentences in which a prepositional object is triggered by the verb. A verb with a prepositional object is the verb *halten* in combination with the preposition *für*. In a sentence like "ich hielt ihn für einen Verrückten", the prepositional phrase *für einen Verrückten*, is an instance of the prepositional object triggered by the verb *halten*.

The FLEX name and description of this column is as follows:

CompPrep (CompPrepLemma) `For␣verbs,␣prepositional␣object␣`

4.3.9 SECOND PREPOSITIONAL OBJECT

The eighth digit in the string of digits is filled in those sentences in which next to the first prepositional object a second prepositional object is triggered by the verb. A verb with a second prepositional object is the verb *herantreten* in combination with the preposition *mit*. In a sentence like "Er trat mit einer Bitte an die Frau heran", the prepositional phrases *mit einer Bitte* and *an die Frau* are instances of the first and the second prepositional object triggered by the verb *herantreten*.

The FLEX name and description of this column is as follows:

CompSecPrep (CompSecPrepLemma) `For␣verbs,␣second␣prepositional␣object␣`

4.3.10 ADVERBIAL COMPLEMENT

The ninth digit in the string of digits is filled in those sentences in which an adverbial complement is triggered by the verb.

Since, apart from the general label 'A' for adverb or prepositional phrase, there are eight different realisations possible for adverbials, we will give an example for all eight forms:

Code	Meaning	Example
A	Adverbial (general)	er flog nach Berlin\zwei Stunden\ in einer Cessna
L	Locative	er wohnt in Kiel
T	Temporal	sie kommen morgen
M	Manner	er geriet außer sich
C	Causative	er weinte vor Schmerz
U	Purpose	er zielt auf Sieg
S	Instrumental	der Vogel flatterte mit den Flügeln
O	Comitative preposition	sie kam zusammen mit ihm
R	Role preposition	er fungiert als Vermittler

Table 20: Example sentences for adverbial complements

The FLEX name and description of this column is as follows:

CompAdv (CompAdvLemma) `For␣verbs,␣adverbial␣complement␣`

4.4 SUBCLASSIFICATION ADJECTIVES

One of the characteristics by which adjectives can be recognized is their gradability. This means that an adjective can be realized in its positive degree, such as the adjective *groß*, or in its comparative degree, such as the form *größer* or in its superlative degree *größt*.

In this column there are four possible values for every adjective:

Code	Meaning	Example
P	non-gradable	übrig
PC	only comparative	ratsam
PS	only superlative	ureigen
PCS	fully gradable	ulkig

Table 21: Codes for gradability of adjectives

For the actual realisations of the inflections it is necessary to consult the wordform lexicon.

The FLEX names and descriptions of these two columns are as follows:

Grad (GradLemma) `For␣adjectives,␣gradability␣`

4.5 SUBCLASSIFICATION NUMERALS

The general term *numerals* covers quantifiers (such as *mehr* or *viel*) and also words which relate directly to numeric values. These 'numeric-value words' can be subdivided into *cardinal* numerals (for example *siebzehn* or *fünftausendsiebenhundertdreiundneunzig*), and *ordinal* numerals (for example *siebzehnte* or *fünftausendsiebenhundertdreiundneunzigste*). The two columns defined here let you distinguish between cardinal and ordinal numerals by means of numeric codes and labels:

Numeric	Label	Example
1	cardinal	acht
2	ordinal	achte
3	fraction	achtel
4	classificatory	achterlei
5	multiplicative	achtfach

Table 22: Codes for numerals

The FLEX names and descriptions of these two columns are as follows:

CardOrdNum (CardOrdNumLemma) `For␣numerals,␣cardinal/ordinal,␣numeric␣`

CardOrd (CardOrdLemma) `For␣numerals,␣cardinal/ordinal,␣labels␣`

4.6 SUBCLASSIFICATION PRONOUNS

There are one hundred and nineteen pronouns given in the D2.5 database, and most of them can be sub-classified in accordance with the codes given in Table 23 (below). The usual numeric codes and labels are available.

Whenever more than one code applies to a particular pronoun, multiple codes are given. For example, the word *wer* can be a relative pronoun, an interrogative pronoun, and an indefinite pronoun. This will be represented by three different entries, each having one code.

Pronoun subclass	Columns		Example
	SubClassPNum	***SubClassP***	
Personal	1	personal	*du*
Demonstrative	2	demonstative	*dieser*
Possessive	3	possessive	*uns*
Relative	4	relative	*der*
Interrogative	5	interrogative	*welcher*
Reflexive	6	reflexive	*sich*
Reciprocal	7	reciprocal	*einander*
Indefinite	8	indefinite	*wenig*

Table 23: Pronoun subclassification codes

The FLEX names and descriptions of these two columns are as follows:

SubClassPNum `For␣pronouns,␣subclasses,␣numeric␣`
(SubClassPNumLemma)

SubClassP `For␣pronouns,␣subclasses,␣labels␣`
(SubClassPLemma)

4.7 SUBCLASSIFICATION PREPOSITIONS

Since German prepositions are able to trigger the case of the noun in the prepositional phrase in which it is embedded, there is a column which gives a numeric code for the particular case triggered by the preposition.

Code	Meaning	Example
2	preposition with genitive	wegen
3	preposition with dative	mit
34	preposition with dative or accusative	an
4	preposition with accusative	durch

Table 24: Code for case triggered by prepositions

The FLEX names and descriptions of these two columns are as follows:

Case (CaseLemma) `For␣prepositions,␣case␣`

5 GERMAN FREQUENCY

The frequency information given in the database (that is, details of how often words occur in German) is available both for lemmas and wordforms. It is taken from the MANNHEIM corpus of the "Institut für deutsche Sprache", which in the 1984 version extracted for CELEX contained about 6.0 million words, taken from written sources of many kinds, and some spoken sources as well. The written sources are texts ranging from highbrow to lowbrow literature, scientific literature, non-specialist literature, memoirs, newspapers and magazines. The spoken sources contain the transcription of "spontaneous speech", which means that the sentences had not in any way been written down or recorded before they were used in conversations, discussions or speeches. Frequency figures are available for lemmas and for wordforms.

The starting point for calculating frequency information is the MANNHEIM 6.0 million word corpus: a count is made of the number of times each string occurs. This task is easy for a computer, which can quickly make a count of all the words that appear in the corpus. The resulting figures are raw 'string' counts – that is, they indicate how many times each separate group of letters occurs in the corpus, taking no account of the different meanings or word classes that can be applied to each group. To develop this basic string count into a more helpful word count, the strings must be identified either as wordforms which can be linked to a particular lemma, or as other things not represented in the database, such as personal names and foreign words.

Sometimes this identification process is straightforward – the string *Bezirken* is only ever the dative plural wordform of the noun lemma *Bezirk*. So in this case the raw string frequency of the string *Bezirken* is also the frequency of the wordform *Bezirken*, and so in the wordform lexicon ***Mann*** column it gets the same frequency as the string.

Once you know the frequencies of the wordforms associated with a particular lemma, working out a frequency figure for the lemma as a whole is straightforward – all you have to do is

add up the appropriate wordform frequencies. In this way the frequency of the noun lemma *Bezirk* is the frequency of the nominative, dative and accusative singular wordform *Bezirk* plus the frequency of the genitive wordform *Bezirks* plus the frequency of the nominative, genitive and accusative plural wordform *Bezirke* plus the frequency of the dative plural wordform *Bezirken*. The frequency of the lemma *Bezirk* is the total of the eight, and this is the figure given in the lemma lexicon ***Mann*** column.

In the following paragraphs we will discuss a way to disambiguate the frequencies of homographic wordforms, such as *Mark* for *coin*, *border* and *marrow*. Although we plan to do the disambiguation as soon as possible, we are not yet able to present the frequencies disambiguated in the way it will be discussed next. However, these paragraphs could not be skipped because otherwise the column ***MannDev***, which is part of the German data of CELEX, would not mean anything to you at the moment, apart from the fact that a figure equal to or greater than the frequency signals a rough split-up of the total string frequency by the number of homographs recognized by CELEX. This is an important point to remember whenever you consult version D2.5 of the German database. It implies that the wordforms *heute* (*today*) and *heute* (*made hay*) were given the same frequency, although this is clearly wide of the mark. This unbalanced frequency distribution is again reflected in the total lemma frequencies. It is not until the release of version D3.0 that this rough split-up will be corrected. Therefore, the figures given below for each of the examples are at best approximations of what the actual figures in D3.0 will look like.

The only way to sort out the individual frequencies of each of a number of homographic strings is to look at the way they are used in the corpus, a process known as *disambiguation*. It's possible to carry out this task quickly by computer program, but at present the results of such programs can never be wholly accurate. For this reason, CELEX chose to disambiguate by hand, which means that someone reads each occurrence of each ambiguous form in the corpus, and notes the lemma to which it belongs. While such an approach is both costly and time-consuming, it does produce results which are more dependable and accurate. For *Messer*, it seems that 84 of the occurrences mean *knife*, and none mean

someone who or something that measures. These are the two figures given in the wordform lexicon ***Mann*** column for the two different *Messer* wordforms. Sometimes not all occurrences refer to wordforms in the database. Some may be proper nouns (surnames, for example) or typing errors, and some simply can't be disambiguated. For example in the corpus *Messer* occurs 12 times in relation to a person's name. Such information is not given in the database since it doesn't relate directly to any of the lemmas or wordforms available.

Again, once the wordform frequencies have been clarified, working out the lemma frequencies is straightforward. For the two lemmas with the form *Messer*, the lemma frequencies are 99 (meaning *knife*), which includes frequencies of 10 for *Messern* and 5 for *Messers*, and 0 (meaning *someone who or something that measures*), giving a total of 99. These lemma frequency figures, which comprise the frequencies of all the other flections of the lemma *Messer* are given in the lemma lexicon ***Mann*** column, and in the same column to be found with the 'lemma information' given for wordforms.

When strings occur very frequently in the corpus, the work required to disambiguate each case by hand can be daunting. It may also be unnecessary, since an intelligent estimate coupled with an indication of how far that estimate is accurate should usually be enough. So, whenever ambiguous words occur more than 100 times in the corpus, not all the occurrences are checked individually. Instead, one hundred occurrences of the string are taken at random from the corpus and then analysed. In this way it's possible to formulate a ratio which indicates the proportions of the various interpretations, and this ratio can then be applied to the real figures to see an estimate of how the fully disambiguated figures would look.

As an example, take the German string *nahe*. Its basic corpus string frequency is 403. It can either be an adjective, a preposition, the first person singular indicative form of the verb *nahen*, the first or third person singular subjunctive form of the verb *nahen* or its imperative singular. Here is a lexicon which shows these wordforms with their word class and frequency:

Word	Class	Mann
nahe	A	153
nahe	PREP	250
nahe	V	24

To calculate these figures, a 100 occurrences of the string *nahe* were taken from the corpus and disambiguated by hand. It turned out that 62 of the occurrences belonged to the preposition lemma, 38 to the adjective lemma and 0 to the verb lemma. So to estimate the real frequency of the wordform belonging to the adjective lemma, divide the number of times it occurred in the sample by the total number of successfully disambiguated forms, and then multiply the result by the original string frequency: $\frac{38}{100} \times 403 = 153$. Repeating this procedure gives 250 for the preposition wordform and 0 for the verb wordform. Displaying just one figure for the verb is the usual way of presenting ambiguous verbal flections, since disambiguating every verbal form by hand is a task which would involve a great deal of work yielding results of interest to only a few.

For most items in the database, the frequency figures are accurate. However, when estimates have to be made on the basis of a hundred examples, then deviation figures have to be calculated, to let you see just how accurate the estimates are. This formula gives the required deviation figure:

$$N \times 1.96 \times \sqrt{\frac{p\,(1-p)}{n} \times \frac{N-n}{N-1}}$$

where N is the frequency of the string as a whole, n is the number of items which could be disambiguated in the random 100-item sample, and p is the ratio figure for the item when it belongs to one particular lemma. Thus for the adjective wordform *nahe*, N is 403, n is 100, and p is 0.38. The formula gives 33.29 as the deviation. This means that the true frequency for this form of *nahe* is almost certain—at least 95% certain—to lie between 120 and 186.

Word	ClassLemma	Mann	MannDev
nahe	A	153	33
nahe	PREP	250	33
nahe	V	0	33

Whenever the deviation is greater than the frequency itself, then you know for sure that some sort of arbitrary approximation has been carried out.

Working out deviation figures for a lemma involves adding together the frequencies of its disambiguated wordforms. And once again, whenever the resulting deviation figure is equal to or greater than the frequency itself, you know that some arbitrary 'disambiguation' has been necessary.

5.1 FREQUENCY INFORMATION FOR LEMMAS AND WORDFORMS

Now that the background details have been explained, the individual column names and descriptions can be formally defined. For both lemmas and wordforms, there are four columns available which express the MANNHEIM frequency figures in various ways.

The first column gives the plain MANNHEIM frequency count for each lemma or wordform. The figure given in the lemma version of the column for *Abänderung* is 17, which means that out of the 6,000,000 words in the corpus, 17 are the word *Abänderung* in some form or other. The figures given in the wordform version of this column reveal how frequently each of the possible forms occur: for *Abänderungen* the figure is 4, for *Abänderung* it is 13. The FLEX name and description of this column are as follows:

Mann `Mannheim␣frequency␣`
(MannLemma)

The second column indicates how accurate the frequencies in the previous column are by providing a deviation figure for each lemma or wordform, calculated according to the methods described in the previous section. If a word has been fully disambiguated without the need for any estimates, the figure is 0. When some estimation has been required, the figure will be greater than zero. If the figure should ever be equal to or greater than the frequency it qualifies, then you know that full disambiguation was not possible. The figure given for the lemma *auf* (as a preposition or an adverb) is 2702, and when you use it in conjunction with the MANNHEIM frequency figure of 39,250 for the preposition, it indicates that you can be almost certain (95% certain)

that the preposition *auf* occurs somewhere between 36,548 and 41,952 times. The FLEX name and description of this column are as follows:

MannDev `Mannheim␣frequency␣deviation␣`
(MannDevLemma)

The next column contains the same frequency figures as the first column, except that they have been scaled down to a range of 1 to 1,000,000 instead of the usual 1 to 6,000,000. This is done by dividing the normal MANNHEIM frequency for each word by the number of words in the whole corpus, and then multiplying the answer by 1,000,000. The end result is a set of figures which are probably easier to understand: it makes greater sense to say that the lemma *Abend* is 133 in a million than it does to say that it's 790 words out of 6,000,000. And since other well-known text corpora—such as the *London-Oslo-Bergen* (LOB) and *Brown* corpora of English—are also based on a count of one million, this scale provides the opportunity for interesting comparisons to be made. However as you might expect, some detail is lost in the scaling-down process: the words *beraten* and *Kritik*, which have the 6.0 million word lemma frequencies of 503 and 507 respectively, both share the same 1 million word frequency of 85.

MannMln `Mannheim␣frequency␣(1,000,000)␣`
(MannMlnLemma)

For those whose work requires a further transformation of the figures (psycholinguists working with stimulus response times for example), a column containing logarithmic values is available. The effect of the logarithmic scale is to emphasize the importance of lower frequency words in a way that the usual linear scale does not. For example, the difference between two words, one of frequency 2 and the other of frequency 1, becomes much greater than the difference between two words of frequency 2002 and 2001. (For the first pair of words, the difference is 0.30103, while for the second pair the difference is a mere 0.000217.) This confirms mathematically what we know intuitively: because there are so many words with a low frequency, the differences between them are that much more significant. With a high frequency word, a difference of one or two isn't very significant.

The values given are the base 10 logarithms of each `Mannheim␣frequency␣(1,000,000)␣` described above. The resulting logarithmic values in this column range from zero ($\log_{10}1$) to 6 ($\log_{10}$1,000,000). And when a word has a normal frequency of zero, the logarithmic value is also given as zero. This is mathematically inaccurate ($\log_x 0$ doesn't exist), but—at least in this context—relatively unimportant: any word with a logarithmic frequency of 0 occurs at the very most only 8 times in the full MANNHEIM 6.0 million word corpus. The thing to remember is that only words which have a MANNHEIM 1,000,000 frequency value of two or more (or, if you prefer, only words which occur 9 or more times in the MANNHEIM corpus) have a logarithmic value greater than zero.

MannLog `Mannheim␣frequency,␣logarithmic␣`
(MannLogLemma)

5.1.1 FREQUENCY INFORMATION FROM WRITTEN AND SPOKEN SOURCES

About 5,400,000 words in the MANNHEIM corpus make up written texts, and the remaining 600,000 words make up spoken texts. In a sense, then, there are two other corpora you can use, one which deals with written texts only and one with spoken texts only. You can choose for yourself whether you wish to use either written or spoken figures in place of the full figures explained in the preceeding sections. The methods used in working out the figures given are the same as those described in the previous section.

The columns available for written and spoken corpus frequencies are roughly the same as those for the full corpus, with the exception of the deviation figures – they are not re-calculated for the written and spoken texts. Instead, you can use the figures given for the full corpus, though remember that when you apply them to frequencies for the written and spoken corpora, the range of error is actually larger than would otherwise be.

5.1.2 WRITTEN CORPUS INFORMATION

There are three columns which contain frequency information for the written sources in the MANNHEIM corpus. The figure given in the lemma version of the column for *Abstand* is

257, which means that out of the 5,400,000 words in the corpus, 257 are the word *Abstand* in some form or other. The figures given in the wordform version of this column reveal how frequently each of the possible forms occur: for *Abstand* the figure is 202, for *Abstände* it is 7, for *Abständen* it is 41, for *Abstande* it is 1, for *Abstandes* it is 2, and for *Abstands* it is 4. The FLEX name and description of this column are as follows:

MannW `Mannheim␣written␣frequency␣5.4m␣`
(MannWLemma)

The next column contains the same frequency figures as ***MannW***, except that they have been scaled down to a range of 1 to 1,000,000 instead of the usual 1 to 5,400,000. This is done by dividing the normal MANNHEIM written frequency for each word by the number of words in the written corpus (about 5,400,000), and then multiplying the answer by 1,000,000. The end result is a set of figures which are probably easier to understand: it makes greater sense to say that a word is one in a million than it does to say that it's 22 words out of 5,400,000. However as you might expect, some detail is lost in the scaling-down process: all words which have 5.4 million word lemma frequencies between 0 and 8 share the same 1 million word frequency of 1.

MannWMln `Mannheim␣written␣frequency␣(1,000,000)␣`
(MannWMlnLemma)

The third and last written corpus column contains the base 10 logarithms of each ***MannWMln***, for the reasons described above in connection with the full corpus. The resulting logarithmic values in this column range from zero ($\log_{10}1$) to 6 ($\log_{10}1{,}000{,}000$). And when a word has a normal frequency of zero, the logarithmic value is also given as zero. This is mathematically inaccurate ($\log_x 0$ doesn't exist), but—at least in this context—relatively unimportant: any word with a logarithmic frequency of 0 occurs at the very most only 8 times in the MANNHEIM 5.4 million written word corpus. The thing to remember is that only words which have a ***MannWMln*** frequency value of two or more (or, if you prefer, only words which occur 9 or more times in the MANNHEIM corpus) have a logarithmic value greater

than zero.

MannWLog `Mannheim␣written␣frequency,␣logarithmic␣`
(MannWLogLemma)

5.1.3 SPOKEN CORPUS INFORMATION

There are three columns which contain frequency information for the spoken sources in the MANNHEIM corpus. The figure given in the lemma version of the column for *Erde* is 60, which means that out of the approximately 600,000 words in the corpus, 60 are the word *Erde* in some form or other. The figures given in the wordform version of this column reveal how frequently each of the possible forms occur: for *Erde* the figure is 59, and for *Erden* it is 1. The FLEX name and description of this column are as follows:

MannS `Mannheim␣spoken␣frequency␣0.6m␣`
(MannSLemma)

The next column contains the same frequency figures as ***MannS***, except that they have been scaled up to a range of 1 to 1,000,000 instead of the usual 1 to 600,000. This is done by dividing the normal MANNHEIM spoken frequency for each word by the number of words in the spoken corpus, and then multiplying the answer by 1,000,000.

MannSMln `Mannheim␣spoken␣frequency␣(1,000,000)␣`
(MannSMlnLemma)

The third and last spoken corpus column contains the base 10 logarithms of each ***MannSMln*** frequency, for the reasons described above in connection with the full corpus. In place of a scale from 1 to 1,000,000, the resulting logarithmic values in this column range from zero ($\log_{10}1$) to 6 ($\log_{10}1{,}000{,}000$). And when a word has a normal frequency of zero, the logarithmic value is also given as zero. This is mathematically inaccurate ($\log_x 0$ doesn't exist), but—at least in this context—relatively unimportant. Because of the extremely small size of the MANNHEIM spoken corpus, every word which occurs once or more has a logarithmic value greater than zero.

MannSLog `Mannheim␣spoken␣frequency,␣logarithmic␣`
(MannSLogLemma)

5.2 FREQUENCY INFORMATION FOR MANNHEIM CORPUS TYPES

The frequency information given in MANNHEIM corpus types lexicons consists of the raw string counts from which all the other frequency figures for lemmas and wordforms are derived. Also available are figures for the spoken and written texts in the corpus for German types which are not to be found amongst the wordforms given in the CELEX database. If you are not already familiar with the terms *token* and *type*, then check the glossary and the first part of the manual, the *Introduction*, in the section 'Lexicon types'.

The first column simply lists the orthographic forms of all types as they occur in the MANNHEIM corpus. The FLEX name and description of this column are as follows:

Type `Graphemic␣transcription␣`

The second column is the basic 'string' count which tells you how many times each type occurs in the MANNHEIM corpus, which contains about 6,000,000 tokens. The FLEX name and description of this column are as follows:

Freq `Absolute␣frequency␣`

To understand the meaning of the third column, you should realize that the MANNHEIM corpus is made up of 316 different texts, which range from complete novels to directions to the use of a cleansing agent for cleaning dentures (Kukident Zahnprothesen-Reinigungs- und Pflegemittel. Gebrauchsanweisung). The figures given here tell you in how many corpus texts each type occurs. For example, *und* occurs in 316 different texts (in fact it occurs in every text in the corpus), *Deutschland* in 129, and *Bier* in 46.

Disp `Dispersion␣`

5.3 FREQUENCY INFORMATION FOR MANNHEIM WRITTEN CORPUS TYPES

The column "Mannheim written frequency" contains raw string counts from the written texts in the MANNHEIM corpus. The FLEX name and description of this column are as follows:

FreqW `Written␣frequency,␣5.4m␣`

The second column shows the dispersion of a word in the written texts of the corpus. For example, the word *Händchenhalten* has a dispersion of 2 over the 316 texts of the entire corpus, since it can only be found in 2 texts of the written part of the corpus. The FLEX name and description of this column are as follows:

DispW `Dispersion␣written␣sources␣`

5.4 FREQUENCY INFORMATION FOR MANNHEIM SPOKEN CORPUS TYPES

The column "Mannheim spoken frequency" contains raw string counts from the spoken texts in the MANNHEIM corpus. About 0.6 million words were transcribed from recorded non-prepared conversations and included in the corpus.

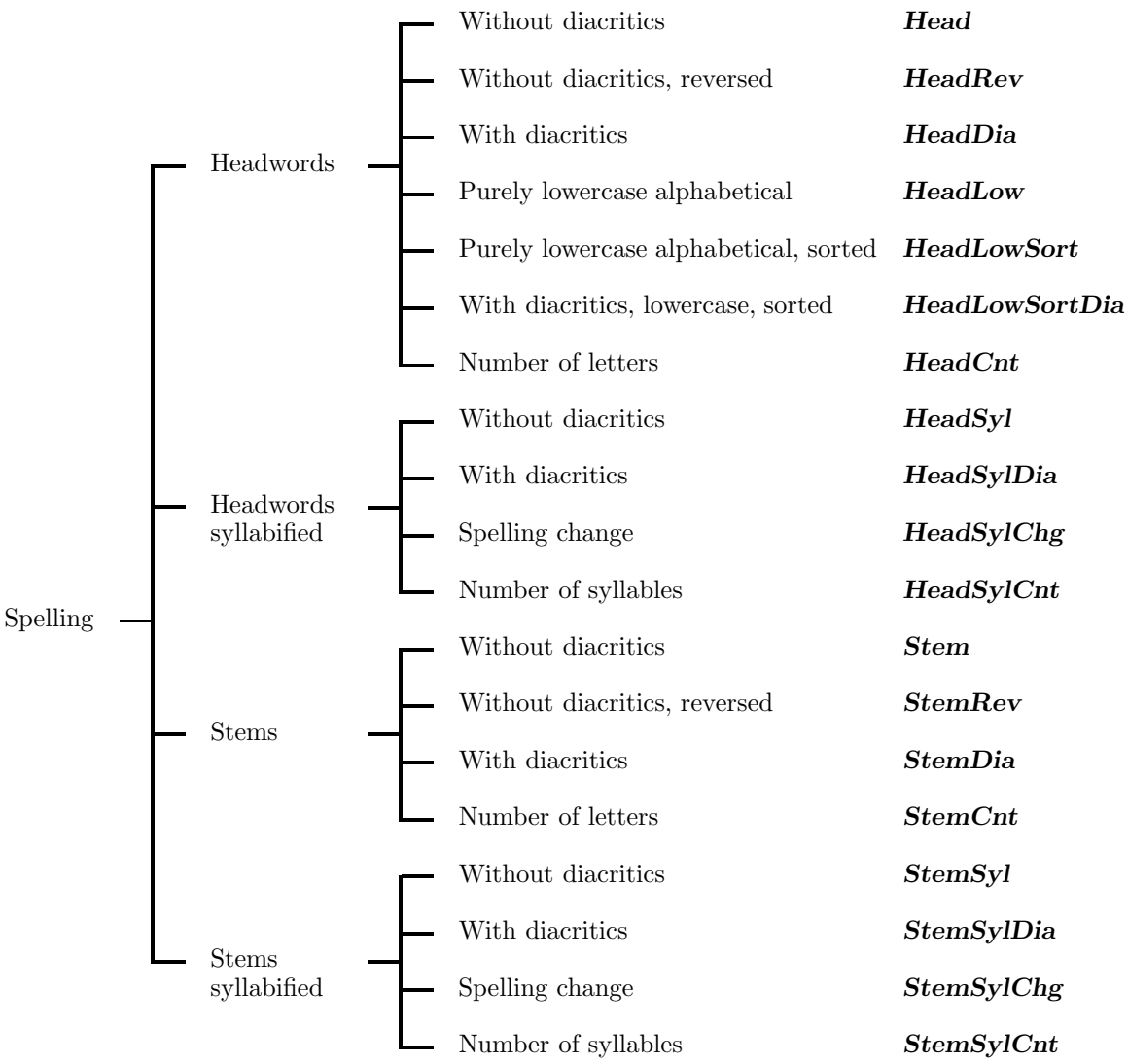
This column contains the frequencies of all types which occur more than once in the spoken texts. The FLEX name and description of this column are as follows:

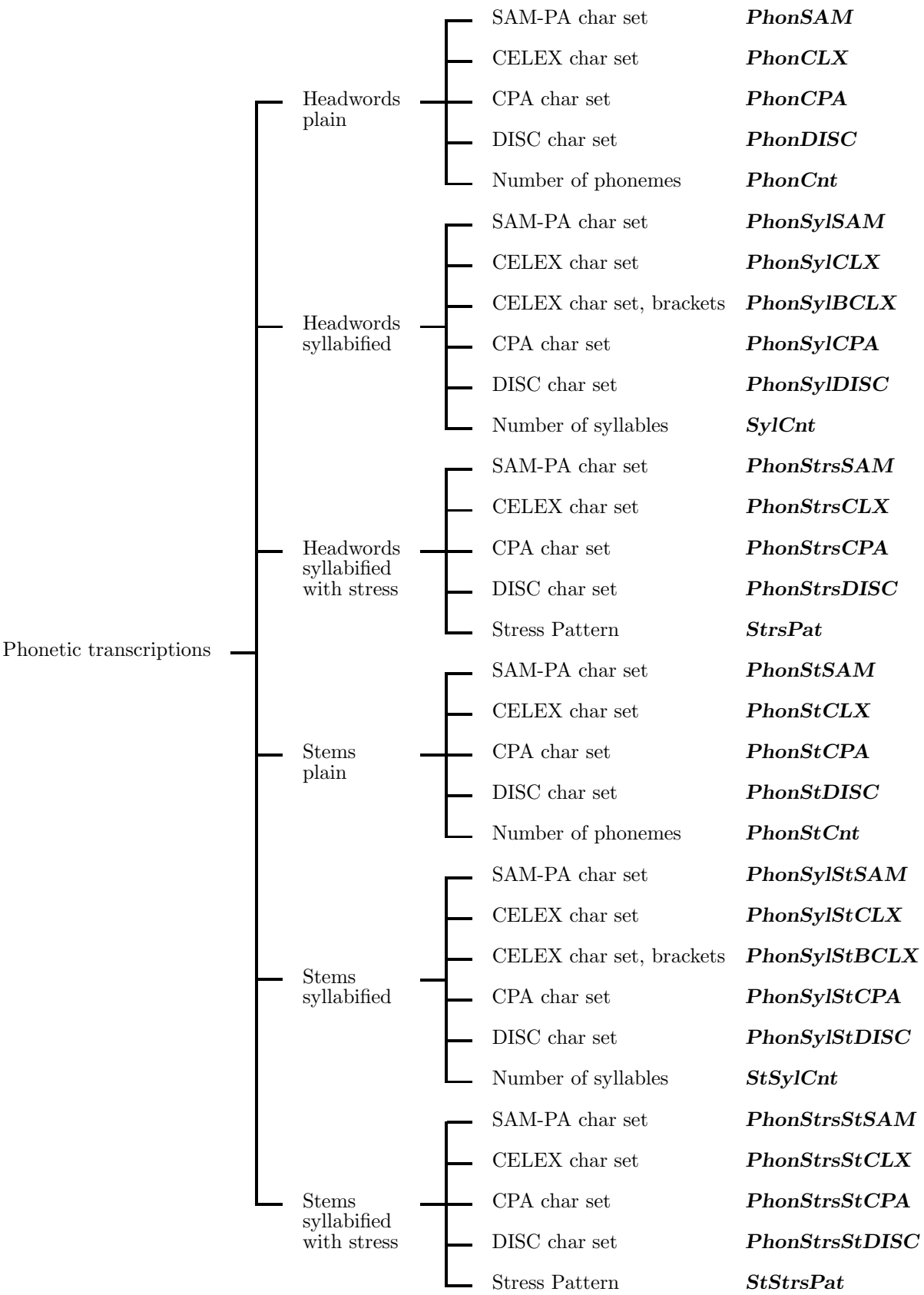
FreqS `Spoken␣frequency,␣0.6m␣`

The second column shows the dispersion of a word in the spoken texts of the corpus. The FLEX name and description of this column are as follows:

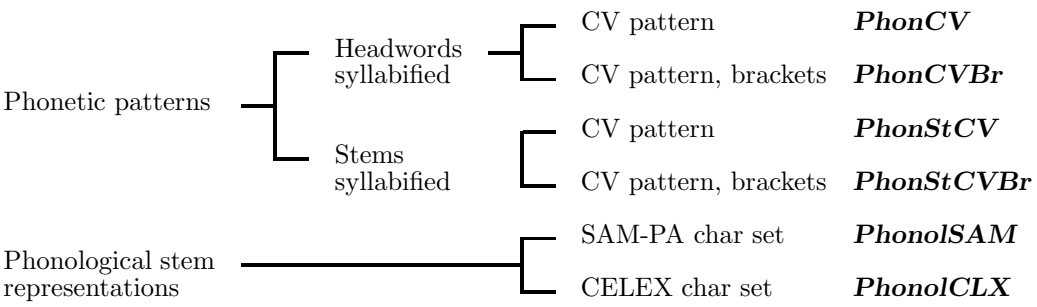
DispS `Dispersion␣spoken␣sources␣`

1 ORTHOGRAPHY OF GERMAN LEMMAS (D25)

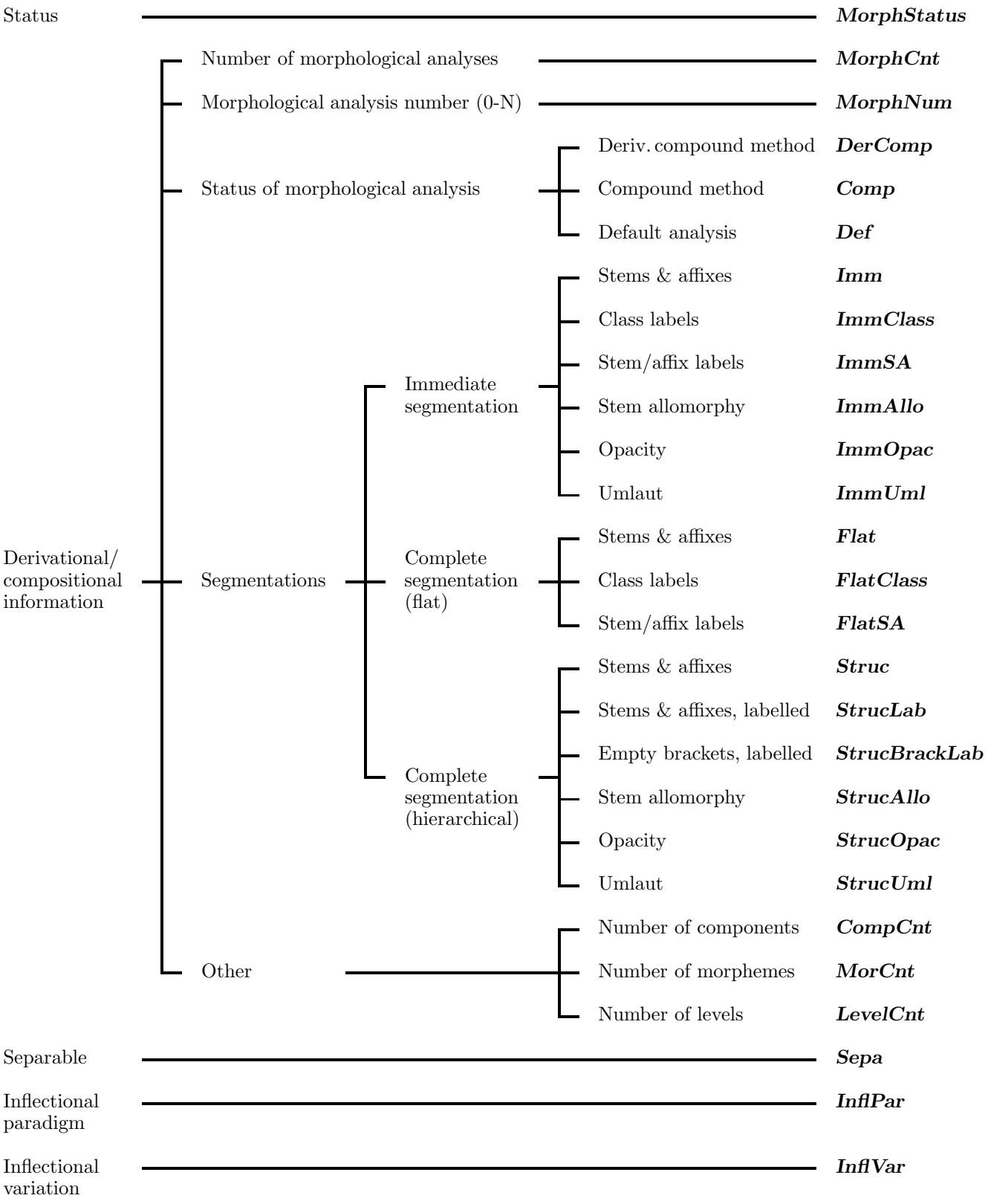



Phonetic transcriptions
Headwords plain
SAM-PA char set
PhonSAM
CELEX char set
PhonCLX
CPA char set
PhonCPA
DISC char set
PhonDISC
Number of phonemes
PhonCnt
Headwords syllabified
SAM-PA char set
PhonSylSAM
CELEX char set
PhonSylCLX
CELEX char set, brackets
PhonSylBCLX
CPA char set
PhonSylCPA
DISC char set
PhonSylDISC
Number of syllables
SylCnt
Headwords syllabified with stress
SAM-PA char set
PhonStrsSAM
CELEX char set
PhonStrsCLX
CPA char set
PhonStrsCPA
DISC char set
PhonStrsDISC
Stress Pattern
StrsPat
Stems plain
SAM-PA char set
PhonStSAM
CELEX char set
PhonStCLX
CPA char set
PhonStCPA
DISC char set
PhonStDISC
Number of phonemes
PhonStCnt
Stems syllabified
SAM-PA char set
PhonSylStSAM
CELEX char set
PhonSylStCLX
CELEX char set, brackets
PhonSylStBCLX
CPA char set
PhonSylStCPA
DISC char set
PhonSylStDISC
Number of syllables
StSylCnt
Stems syllabified with stress
SAM-PA char set
PhonStrsStSAM
CELEX char set
PhonStrsStCLX
CPA char set
PhonStrsStCPA
DISC char set
PhonStrsStDISC
Stress Pattern
StStrsPat

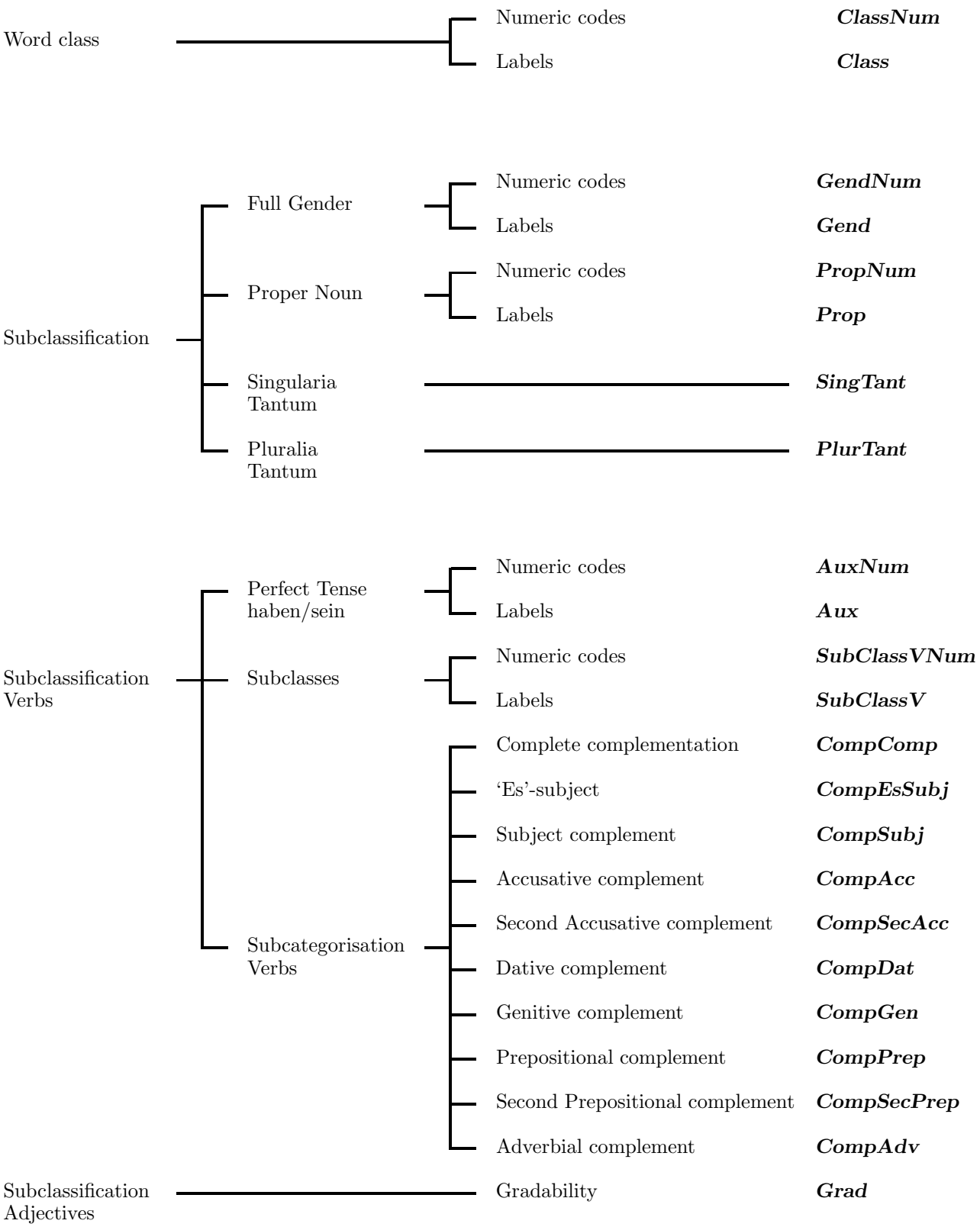
3 PHONOLOGY OF GERMAN LEMMAS (D25)



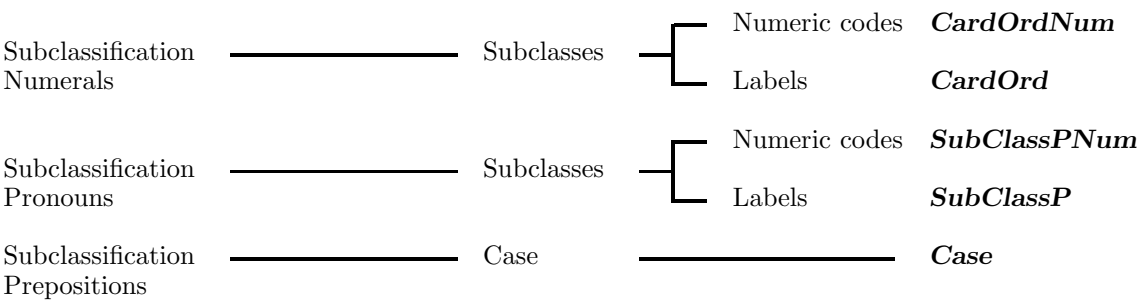
4 MORPHOLOGY OF GERMAN LEMMAS (D25)



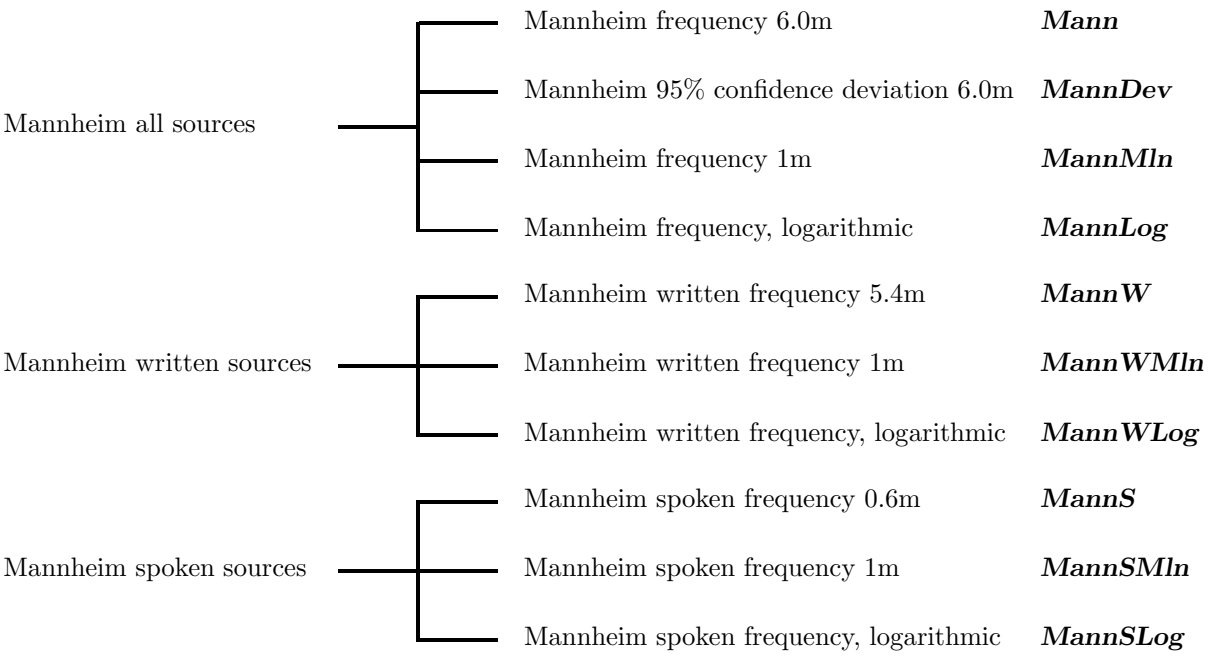
5 SYNTAX OF GERMAN LEMMAS (D25)



6 SYNTAX OF GERMAN LEMMAS (D25)



7 FREQUENCY OF GERMAN LEMMAS (D25)



Aux	For␣verbs:␣auxiliary␣verb,␣labels␣

Type:	character␣	Null values:	0␣
Minimum value:	haben␣	Minimum length:	4␣
Maximum value:	sein␣	Maximum length:	10␣
Characters:	/ a b e h i n s␣		

AuxNum	For␣verbs:␣auxiliary␣verb,␣numeric␣

Type:	character␣	Null values:	0␣
Minimum value:	1␣	Minimum length:	1␣
Maximum value:	2␣	Maximum length:	2␣
Characters:	1 2␣		

CardOrd	For␣numerals:␣subclasses,␣labels␣

Type:	character␣	Null values:	51599␣
Minimum value:	cardinal␣	Minimum length:	7␣
Maximum value:	ordinal␣	Maximum length:	14␣
Characters:	a c d e f g i l m n o p r t u v␣		

CardOrdNum	For␣numerals:␣subclasses,␣numeric␣

Type:	character␣	Null values:	51599␣
Minimum value:	1␣	Minimum length:	1␣
Maximum value:	5␣	Maximum length:	1␣
Characters:	1 2 3 4 5␣		

Case	For␣prepositions:␣case␣

Type:	character␣	Null values:	51621␣
Minimum value:	2␣	Minimum length:	1␣
Maximum value:	4␣	Maximum length:	2␣
Characters:	2 3 4␣		

Class	Word␣class,␣labels␣		
Type:	character␣	Null values:	0␣
Minimum value:	A␣	Minimum length:	1␣
Maximum value:	V␣	Maximum length:	4␣
Characters:	A␣C␣D␣E␣I␣M␣N␣O␣P␣R␣T␣U␣V␣		

ClassNum	Word␣class,␣numeric␣		
Type:	character␣	Null values:	0␣
Minimum value:	1␣	Minimum length:	1␣
Maximum value:	10␣	Maximum length:	2␣
Characters:	0␣1␣2␣3␣4␣5␣6␣7␣8␣9␣		

Comp	Compound␣analysis␣method␣		
Type:	character␣	Null values:	0␣
Minimum value:	N␣	Minimum length:	1␣
Maximum value:	Y␣	Maximum length:	1␣
Characters:	N␣Y␣		

CompAcc	For␣verbs:␣Accusative␣complement␣		
Type:	character␣	Null values:	42333␣
Minimum value:	I␣	Minimum length:	1␣
Maximum value:	U␣	Maximum length:	1␣
Characters:	I␣O␣P␣U␣		

CompAdv	For␣verbs:␣Adverbial␣complement␣		
Type:	character␣	Null values:	42333␣
Minimum value:	I␣	Minimum length:	1␣
Maximum value:	U␣	Maximum length:	1␣
Characters:	I␣O␣P␣U␣		

CompCnt	Number␣of␣morphological␣components␣

Type:	numeric␣	Null values:	0␣
Minimum value:	1␣	Minimum length:	1␣
Maximum value:	4␣	Maximum length:	1␣
Characters:	0␣1␣2␣3␣4␣		

CompComp	For␣verbs:␣complete␣segmentation␣

Type:	character␣	Null values:	42333␣
Minimum value:	000000000;␣	Minimum length:	10␣
Maximum value:	EG0000000;0M0000000;␣ 0G0000000;0000N0000;␣ 00000N000;000000000;␣	Maximum length:	160␣
Characters:	0␣;␣?␣A␣C␣E␣G␣I␣L␣M␣N␣O␣P␣R␣S␣T␣U␣Z␣i␣n␣p␣z␣		

CompDat	For␣verbs:␣Dative␣complement␣

Type:	character␣	Null values:	42333␣
Minimum value:	I␣	Minimum length:	1␣
Maximum value:	U␣	Maximum length:	1␣
Characters:	I␣O␣P␣U␣		

CompEsSubj	For␣verbs:␣'Es'-Subject␣complement␣

Type:	character␣	Null values:	42333␣
Minimum value:	I␣	Minimum length:	1␣
Maximum value:	U␣	Maximum length:	1␣
Characters:	I␣O␣P␣U␣		

CompGen	For␣verbs:␣Genitive␣complement␣

Type:	character␣	Null values:	42333␣
Minimum value:	I␣	Minimum length:	1␣
Maximum value:	U␣	Maximum length:	1␣
Characters:	I O P U␣		

CompPrep	For␣verbs:␣Prepositional␣complement␣

Type:	character␣	Null values:	42333␣
Minimum value:	I␣	Minimum length:	1␣
Maximum value:	U␣	Maximum length:	1␣
Characters:	I O P U␣		

CompSecAcc	For␣verbs:␣Second␣Accusative␣complement␣

Type:	character␣	Null values:	42333␣
Minimum value:	I␣	Minimum length:	1␣
Maximum value:	U␣	Maximum length:	1␣
Characters:	I O P U␣		

CompSecPrep	For␣verbs:␣Second␣Prepositional␣complement␣

Type:	character␣	Null values:	42333␣
Minimum value:	I␣	Minimum length:	1␣
Maximum value:	U␣	Maximum length:	1␣
Characters:	I O P U␣		

CompSubj	For␣verbs:␣Subject␣complement␣

Type:	character␣	Null values:	42333␣
Minimum value:	I␣	Minimum length:	1␣
Maximum value:	U␣	Maximum length:	1␣
Characters:	I O P U␣		

Def	Default␣analysis␣

Type:	character␣	Null values:	0␣
Minimum value:	N␣	Minimum length:	1␣
Maximum value:	Y␣	Maximum length:	1␣
Characters:	N␣Y␣		

DerComp	Derivational␣compound␣analysis␣method␣

Type:	character␣	Null values:	0␣
Minimum value:	N␣	Minimum length:	1␣
Maximum value:	Y␣	Maximum length:	1␣
Characters:	N␣Y␣		

Flat	Flat␣segmentation␣

Type:	character␣	Null values:	0␣
Minimum value:	A␣	Minimum length:	1␣
Maximum value:	zytogen␣	Maximum length:	39␣
Characters:	+␣A␣B␣C␣D␣E␣F␣G␣H␣I␣J␣K␣L␣M␣N␣O␣P␣Q␣R␣S␣T␣U␣V␣ W␣X␣Y␣Z␣a␣b␣c␣d␣e␣f␣g␣h␣i␣j␣k␣l␣m␣n␣o␣p␣q␣r␣s␣ t␣u␣v␣w␣x␣y␣z␣		

FlatClass	Flat␣segmentation,␣word␣class␣labels␣

Type:	character␣	Null values:	0␣
Minimum value:	A␣	Minimum length:	1␣
Maximum value:	xxxVxx␣	Maximum length:	9␣
Characters:	A␣B␣C␣D␣F␣I␣N␣O␣P␣Q␣R␣V␣n␣x␣		

FlatSA Flat segmentation, stem/affix labels

Type:	character	Null values:	0
Minimum value:	A	Minimum length:	1
Maximum value:	nSA	Maximum length:	9
Characters:	A S		

Gend For nouns: gender, labels

Type:	character	Null values:	21000
Minimum value:	F	Minimum length:	1
Maximum value:	NM	Maximum length:	3
Characters:	F M N		

GendNum For nouns: gender, numeric

Type:	character	Null values:	21000
Minimum value:	1	Minimum length:	1
Maximum value:	32	Maximum length:	3
Characters:	1 2 3		

Grad For adjectives: gradability

Type:	character	Null values:	41873
Minimum value:	P	Minimum length:	1
Maximum value:	PS	Maximum length:	3
Characters:	C P S		

Head	Headword␣

Type:	character␣	Null values:	0␣
Minimum value:	A␣	Minimum length:	1␣
Maximum value:	zytogen␣	Maximum length:	31␣
Characters:	A␣B␣C␣D␣E␣F␣G␣H␣I␣J␣K␣L␣M␣N␣O␣P␣Q␣R␣S␣T␣U␣V␣ W␣X␣Y␣Z␣a␣b␣c␣d␣e␣f␣g␣h␣i␣j␣k␣l␣m␣n␣o␣p␣q␣r␣s␣ t␣u␣v␣w␣x␣y␣z␣		

HeadCnt	Headword,number␣of␣letters␣

Type:	numeric␣	Null values:	0␣
Minimum value:	1␣	Minimum length:	1␣
Maximum value:	31␣	Maximum length:	2␣
Characters:	0␣1␣2␣3␣4␣5␣6␣7␣8␣9␣		

HeadDia	Headword,␣diacritics␣

Type:	character␣	Null values:	0␣
Minimum value:	A␣	Minimum length:	1␣
Maximum value:	üppig␣	Maximum length:	31␣
Characters:	Ä␣Ö␣Ü␣ß␣ä␣é␣ö␣ü␣A␣B␣C␣D␣E␣F␣G␣H␣I␣J␣K␣L␣M␣N␣O␣ P␣Q␣R␣S␣T␣U␣V␣W␣X␣Y␣Z␣a␣b␣c␣d␣e␣f␣g␣h␣i␣j␣k␣l␣ m␣n␣o␣p␣q␣r␣s␣t␣u␣v␣w␣x␣y␣z␣		

HeadLow	Headword,␣lowercase,␣alphabetical␣

Type:	character␣	Null values:	0␣
Minimum value:	a␣	Minimum length:	1␣
Maximum value:	zytostom␣	Maximum length:	31␣
Characters:	a␣b␣c␣d␣e␣f␣g␣h␣i␣j␣k␣l␣m␣n␣o␣p␣q␣r␣s␣t␣u␣v␣w␣ x␣y␣z␣		

HeadLowSort	Headword, lowercas, alphabetical, sorted

Type:	caharacter	Null values:	0
Minimum value:	a	Minimum length:	1
Maximum value:	z	Maximum length:	31
Characters:	a b c d e f g h i j k l m n o p q r s t u v w x y z		

HeadLowSortDia	Headword, lowercase, sorted, diacritics

Type:	character	Null values:	0
Minimum value:	a	Minimum length:	1
Maximum value:	ü	Maximum length:	31
Characters:	ß ä é ö ü a b c d e f g h i j k l m n o p q r s t u v w x y z		

HeadRev	Headword, reversed

Type:	character	Null values:	0
Minimum value:	A	Minimum length:	1
Maximum value:	zzaJ	Maximum length:	31
Characters:	A B C D E F G H I J K L M N O P Q R S T U V W X Y Z a b c d e f g h i j k l m n o p q r s t u v w x y z		

HeadSyl	Headword, syllabified

Type:	character	Null values:	0
Minimum value:	A	Minimum length:	1
Maximum value:	zy-to-gen	Maximum length:	40
Characters:	- = A B C D E F G H I J K L M N O P Q R S T U V W X Y Z a b c d e f g h i j k l m n o p q r s t u v w x y z		

HeadSylChg	Spelling␣change,␣headword␣

Type:	character␣	Null values:	0␣
Minimum value:	N␣	Minimum length:	1␣
Maximum value:	Y␣	Maximum length:	1␣
Characters:	N␣Y␣		

HeadSylCnt	Headword,␣number␣of␣orthographic␣syllables␣

Type:	numeric␣	Null values:	0␣
Minimum value:	1␣	Minimum length:	1␣
Maximum value:	10␣	Maximum length:	2␣
Characters:	0␣1␣2␣3␣4␣5␣6␣7␣8␣9␣		

HeadSylDia	Headword,␣syllabified,␣diacritics␣

Type:	character␣	Null values:	0␣
Minimum value:	A␣	Minimum length:	1␣
Maximum value:	üp-pig␣	Maximum length:	40␣
Characters:	Ä␣Ö␣Ü␣ß␣ä␣é␣ö␣ü␣- =␣A␣B␣C␣D␣E␣F␣G␣H␣I␣J␣K␣L␣M␣ N␣O␣P␣Q␣R␣S␣T␣U␣V␣W␣X␣Y␣Z␣a␣b␣c␣d␣e␣f␣g␣h␣i␣j␣ k␣l␣m␣n␣o␣p␣q␣r␣s␣t␣u␣v␣w␣x␣y␣z␣		

IdNum	Lemma␣number␣

Type:	numeric␣	Null values:	0␣
Minimum value:	1␣	Minimum length:	1␣
Maximum value:	51682␣	Maximum length:	5␣
Characters:	0␣1␣2␣3␣4␣5␣6␣7␣8␣9␣		

Imm	Immediate␣segmentation␣

Type:	character␣	Null values:	0␣
Minimum value:	A␣	Minimum length:	1␣
Maximum value:	zytogen␣	Maximum length:	33␣
Characters:	+␣A␣B␣C␣D␣E␣F␣G␣H␣I␣J␣K␣L␣M␣N␣O␣P␣Q␣R␣S␣T␣U␣ V␣W␣X␣Y␣Z␣a␣b␣c␣d␣e␣f␣g␣h␣i␣j␣k␣l␣m␣n␣o␣p␣q␣r␣ s␣t␣u␣v␣w␣x␣y␣z␣		

ImmAllo	Stem␣allomorphy,␣top␣level␣

Type:	character␣	Null values:	0␣
Minimum value:	N␣	Minimum length:	1␣
Maximum value:	Y␣	Maximum length:	1␣
Characters:	N␣Y␣		

ImmClass	Immediate␣segmentation,␣word␣class␣labels␣

Type:	character␣	Null values:	0␣
Minimum value:	A␣	Minimum length:	1␣
Maximum value:	xxN␣	Maximum length:	4␣
Characters:	A␣B␣C␣D␣F␣I␣N␣O␣P␣Q␣R␣V␣c␣n␣p␣x␣		

ImmOpac	Opacity,␣top␣level␣

Type:	character␣	Null values:	0␣
Minimum value:	N␣	Minimum length:	1␣
Maximum value:	Y␣	Maximum length:	1␣
Characters:	N␣Y␣		

ImmSA	Immediate segmentation, stem/affix labels

Type:	character	Null values:	0
Minimum value:	A	Minimum length:	1
Maximum value:	SSS	Maximum length:	4
Characters:	A S		

ImmUml	Umlaut, top level

Type:	character	Null values:	0
Minimum value:	N	Minimum length:	1
Maximum value:	Y	Maximum length:	1
Characters:	N Y		

InflPar	Inflectional paradigm

Type:	character	Null values:	0
Minimum value:	A	Minimum length:	1
Maximum value:	r6	Maximum length:	7
Characters:	/ 0 1 2 3 4 5 6 7 8 9 A I P S U i r		

InflVar	Inflectional variation

Type:	character	Null values:	0
Minimum value:	N	Minimum length:	1
Maximum value:	Y	Maximum length:	1
Characters:	N Y		

LevelCnt	Number of morphological levels

Type:	numeric	Null values:	0
Minimum value:	0	Minimum length:	1
Maximum value:	7	Maximum length:	1
Characters:	0 1 2 3 4 5 6 7		

MannDev	`Mannheim␣frequency␣deviation␣`

Type:	`numeric␣`	Null values:	`0␣`
Minimum value:	`0␣`	Minimum length:	`1␣`
Maximum value:	`438972␣`	Maximum length:	`6␣`
Characters:	`0 1 2 3 4 5 6 7 8 9␣`		

MannLog	`Mannheim␣frequency,␣logarithmic␣`

Type:	`numeric␣`	Null values:	`0␣`
Minimum value:	`0␣`	Minimum length:	`1␣`
Maximum value:	`4.5682␣`	Maximum length:	`6␣`
Characters:	`. 0 1 2 3 4 5 6 7 8 9␣`		

MannMln	`Mannheim␣frequency␣(1,000,000)␣`

Type:	`numeric␣`	Null values:	`0␣`
Minimum value:	`0␣`	Minimum length:	`1␣`
Maximum value:	`36996␣`	Maximum length:	`5␣`
Characters:	`0 1 2 3 4 5 6 7 8 9␣`		

MannS	`Mannheim␣spoken␣frequency␣0.6m␣`

Type:	`numeric␣`	Null values:	`0␣`
Minimum value:	`0␣`	Minimum length:	`1␣`
Maximum value:	`18736␣`	Maximum length:	`5␣`
Characters:	`0 1 2 3 4 5 6 7 8 9␣`		

MannSLog	`Mannheim␣spoken␣frequency,␣logarithmic␣`

Type:	`numeric␣`	Null values:	`0␣`
Minimum value:	`0␣`	Minimum length:	`1␣`
Maximum value:	`4.5039␣`	Maximum length:	`6␣`
Characters:	`. 0 1 2 3 4 5 6 7 8 9␣`		

MannSMln	Mannheim␣spoken␣frequency␣(1,000,000)␣

Type:	numeric␣	Null values:	0␣
Minimum value:	0␣	Minimum length:	1␣
Maximum value:	31908␣	Maximum length:	5␣
Characters:	0 1 2 3 4 5 6 7 8 9␣		

MannW	Mannheim␣written␣frequency␣5.4m␣

Type:	numeric␣	Null values:	0␣
Minimum value:	0␣	Minimum length:	1␣
Maximum value:	201462␣	Maximum length:	6␣
Characters:	0 1 2 3 4 5 6 7 8 9␣		

MannWLog	Mannheim␣written␣frequency,␣logarithmic␣

Type:	numeric␣	Null values:	0␣
Minimum value:	0␣	Minimum length:	1␣
Maximum value:	4.5746␣	Maximum length:	6␣
Characters:	. 0 1 2 3 4 5 6 7 8 9␣		

MannWMln	Mannheim␣written␣frequency␣(1,000,000)␣

Type:	numeric␣	Null values:	0␣
Minimum value:	0␣	Minimum length:	1␣
Maximum value:	37553␣	Maximum length:	5␣
Characters:	0 1 2 3 4 5 6 7 8 9␣		

MorCnt	Number␣of␣morphemes␣

Type:	numeric␣	Null values:	0␣
Minimum value:	0␣	Minimum length:	1␣
Maximum value:	9␣	Maximum length:	1␣
Characters:	0 1 2 3 4 5 6 7 8 9␣		

MorphCnt	Number␣of␣morphological␣analyses␣

Type:	numeric␣	Null values:	0␣
Minimum value:	0␣	Minimum length:	1␣
Maximum value:	9␣	Maximum length:	1␣
Characters:	0␣1␣2␣3␣4␣5␣6␣7␣8␣9␣		

MorphNum	Morphological␣analysis␣number␣

Type:	numeric␣	Null values:	0␣
Minimum value:	1␣	Minimum length:	1␣
Maximum value:	3␣	Maximum length:	1␣
Characters:	0␣1␣2␣3␣		

MorphStatus	Morphological␣Status␣

Type:	character␣	Null values:	0␣
Minimum value:	C␣	Minimum length:	1␣
Maximum value:	Z␣	Maximum length:	1␣
Characters:	C␣F␣I␣M␣U␣Z␣		

PhonCLX	Phon.␣headword,␣CELEX␣charset␣

Type:	character␣	Null values:	0␣
Minimum value:	&:.␣	Minimum length:	3␣
Maximum value:	z.y:.t.z.y:.t.O.s.t.␣	Maximum length:	57␣
Characters:	&␣.␣3␣:␣@␣A␣E␣I␣N␣O␣Q␣S␣U␣V␣Y␣Z␣a␣b␣d␣e␣f␣g␣h␣ i␣j␣k␣l␣m␣n␣o␣p␣r␣s␣t␣u␣v␣w␣x␣y␣z␣~␣		

PhonCnt	Headword,number␣of␣phonemes␣

Type:	numeric␣	Null values:	0␣
Minimum value:	1␣	Minimum length:	1␣
Maximum value:	27␣	Maximum length:	2␣
Characters:	0␣1␣2␣3␣4␣5␣6␣7␣8␣9␣		

PhonCPA	Phon.␣headword,␣CPA␣charset␣

Type:	character␣	Null values:	0␣
Minimum value:	@.d.v.A:.n.t.I.J/.␣	Minimum length:	3␣
Maximum value:	z.y:.t.z.y:.t.O.s.t.␣	Maximum length:	57␣
Characters:	.␣/␣:␣@␣A␣C␣E␣I␣J␣N␣O␣Q␣S␣T␣U␣Y␣Z␣ˆ␣a␣b␣d␣e␣f␣ g␣h␣i␣j␣k␣l␣m␣n␣o␣p␣q␣r␣s␣t␣u␣v␣w␣x␣y␣z␣˜␣		

PhonCV	Headword,␣phon.␣CV␣pattern␣

Type:	character␣	Null values:	0␣
Minimum value:	CCCVC␣	Minimum length:	2␣
Maximum value:	VVCCC-VVC-CVV-CVC␣	Maximum length:	40␣
Characters:	- C␣V␣		

PhonCVBr	Headword,␣phon.␣CV␣pattern,␣with␣brackets␣

Type:	character␣	Null values:	0␣
Minimum value:	[CCCVCC]␣	Minimum length:	4␣
Maximum value:	[V] [VV] [CVV] [CVV] [VC]␣	Maximum length:	50␣
Characters:	C␣V␣[␣]␣		

PhonDISC Phon. headword, DISC charset

Type: character — Null values: 0
Minimum value: $lr6ndSpOrtl@r — Minimum length: 1
Maximum value: |z@ — Maximum length: 27
Characters: # $ &) + / 0 1 2 3 4 6 = @ A B E I J N O S U V W X Y Z ^ _ a b c d e f g h i j k l m n o p q r s t u v w x y z { | ~

PhonolCLX Phonological deep structure, CELEX charset

Type: character — Null values: 3792
Minimum value: &: — Minimum length: 2
Maximum value: zy:s+Ixkait — Maximum length: 35
Characters: # & + : @ A E I N O S U Y Z a b d e f g h i j k l m n o p r s t u v x y z { | ~

PhonolSAM Phonological deep structure, SAM-PA charset

Type: character — Null values: 3792
Minimum value: /rt@r — Minimum length: 2
Maximum value: |:z@ — Maximum length: 35
Characters: # + / : @ A E I N O S U Y Z a b d e f g h i j k l m n o p r s t u v x y z { | ~

PhonSAM Phon. headword, SAM-PA charset

Type: character — Null values: 0
Minimum value: /.f.@.n.t.l.I.x. — Minimum length: 3
Maximum value: |:.z.@. — Maximum length: 57
Characters: . / 3 : @ A E I N O S U V Y Z a b d e f g h i j k l m n o p r s t u v w x y z { | ~

PhonStCLX Phon. stem, CELEX charset

Type:	character	Null values:	0
Minimum value:	&:.	Minimum length:	3
Maximum value:	z.y:.t.z.y:.t.O.s.t.	Maximum length:	57
Characters:	& . 3 : @ A E I N O Q S U V Y Z a b d e f g h i j k l m n o p r s t u v w x y z ~		

PhonStCnt Stem, number of phonemes

Type:	numeric	Null values:	0
Minimum value:	1	Minimum length:	1
Maximum value:	27	Maximum length:	2
Characters:	0 1 2 3 4 5 6 7 8 9		

PhonStCPA Phon. stem, CPA charset

Type:	character	Null values:	0
Minimum value:	@.d.v.A:.n.t.I.J/.	Minimum length:	3
Maximum value:	z.y:.t.z.y:.t.O.s.t.	Maximum length:	57
Characters:	. / : @ A C E I J N O Q S T U Y Z ^ a b d e f g h i j k l m n o p q r s t u v w x y z ~		

PhonStCV Stem, phon. CV pattern

Type:	character	Null values:	0
Minimum value:	CCCVC	Minimum length:	2
Maximum value:	VVCCC-VVC-CVVC	Maximum length:	40
Characters:	- C V		

PhonStCVBr Stem, phon. CV pattern, with brackets

Type:	character	Null values:	0
Minimum value:	[CCCVCC]	Minimum length:	4
Maximum value:	[V] [VV] [CVV] [CVV] [VC]	Maximum length:	50
Characters:	C V [ ]		

PhonStDISC Phon. stem, DISC charset

Type:	character	Null values:	0
Minimum value:	$lr6ndSpOrtl@r	Minimum length:	1
Maximum value:	|z@	Maximum length:	27
Characters:	# $ &) + / 0 1 2 3 4 6 = @ A B E I J N O S U V W X Y Z ^ _ a b c d e f g h i j k l m n o p q r s t u v w x y z { | ~		

PhonStrsCLX Syll. phon. headword, with stress, CELEX charset

Type:	character	Null values:	0
Minimum value:	&:-di:-'pa:l	Minimum length:	3
Maximum value:	zy:t-zy:t-'Ost	Maximum length:	42
Characters:	" & ' - 3 : @ A E I N O Q S U V Y Z a b d e f g h i j k l m n o p r s t u v w x y z ~		

PhonStrsCPA Syll. phon. headword, with stress, CPA charset

Type:	character	Null values:	0
Minimum value:	'A/	Minimum length:	3
Maximum value:	zy:t.zy:t.'Ost	Maximum length:	42
Characters:	" ' . / : @ A C E I J N O Q S T U Y Z ^ a b d e f g h i j k l m n o p q r s t u v w x y z ~		

PhonStrsDISC	Syll.␣phon.␣headword,␣with␣stress,␣DISC␣charset␣

Type:	character␣	Null values:	0␣
Minimum value:	&-’=a-li-@␣	Minimum length:	2␣
Maximum value:	|-ku-’me-nIS␣	Maximum length:	37␣
Characters:	"␣#␣$␣&␣’␣)␣+␣- /␣0␣1␣2␣3␣4␣6␣=␣@␣A␣B␣E␣I␣J␣N␣ O␣S␣U␣V␣W␣X␣Y␣Z␣^␣_␣a␣b␣c␣d␣e␣f␣g␣h␣i␣j␣k␣l␣m␣ n␣o␣p␣q␣r␣s␣t␣u␣v␣w␣x␣y␣z␣{␣|␣~␣		

PhonStrsSAM	Syll.␣phon.␣headword,␣with␣stress,SAM-PA␣charset␣

Type:	character␣	Null values:	0␣
Minimum value:	"/-f@nt-lIx␣	Minimum length:	3␣
Maximum value:	|:-ku:-"me:-nIS␣	Maximum length:	42␣
Characters:	"␣%␣- /␣3␣:␣@␣A␣E␣I␣N␣O␣S␣U␣V␣Y␣Z␣a␣b␣d␣e␣f␣g␣ h␣i␣j␣k␣l␣m␣n␣o␣p␣r␣s␣t␣u␣v␣w␣x␣y␣z␣{␣|␣~␣		

PhonStrsStCLX	Syll.␣phon.␣stem,␣with␣stress,␣CELEX␣charset␣

Type:	character␣	Null values:	0␣
Minimum value:	&:-di:-’pa:l␣	Minimum length:	3␣
Maximum value:	zy:t-zy:t-’Ost␣	Maximum length:	42␣
Characters:	&␣’␣- 3␣:␣@␣A␣E␣I␣N␣O␣Q␣S␣U␣V␣Y␣Z␣a␣b␣d␣e␣f␣g␣ h␣i␣j␣k␣l␣m␣n␣o␣p␣r␣s␣t␣u␣v␣w␣x␣y␣z␣~␣		

PhonStrsStCPA	Syll.␣phon.␣stem,␣with␣stress,␣CPA␣charset␣

Type:	character␣	Null values:	0␣
Minimum value:	’A/␣	Minimum length:	3␣
Maximum value:	zy:t.zy:t.’Ost␣	Maximum length:	42␣
Characters:	’␣.␣/␣:␣@␣A␣C␣E␣I␣J␣N␣O␣Q␣S␣T␣U␣Y␣Z␣^␣a␣b␣d␣e␣ f␣g␣h␣i␣j␣k␣l␣m␣n␣o␣p␣q␣r␣s␣t␣u␣v␣w␣x␣y␣z␣~␣		

PhonStrsStDISC

Syll. phon. stem, with stress, DISC charset

Type:	character	Null values:	0
Minimum value:	&-'=a-li-@	Minimum length:	2
Maximum value:	|-ku-'me-nIS	Maximum length:	37
Characters:	# $ & ') + - / 0 1 2 3 4 6 = @ A B E I J N O S U V W X Y Z ^ _ a b c d e f g h i j k l m n o p q r s t u v w x y z { | ~			

PhonStrsStSAM

Syll. phon. stem, with stress, SAM-PA charset

Type:	character	Null values:	0
Minimum value:	"/-f@nt-lIx	Minimum length:	3
Maximum value:	|:-ku:-"me:-nIS	Maximum length:	42
Characters:	" - / 3 : @ A E I N O S U V Y Z a b d e f g h i j k l m n o p r s t u v w x y z { | ~			

PhonStSAM

Phon. stem, SAM-PA charset

Type:	character	Null values:	0
Minimum value:	/.f.@.n.t.l.I.x.	Minimum length:	3
Maximum value:	|:.z.@.	Maximum length:	57
Characters:	. / 3 : @ A E I N O S U V Y Z a b d e f g h i j k l m n o p r s t u v w x y z { | ~			

PhonSylBCLX

Syll. phon. headword, CELEX charset (brackets)

Type:	character	Null values:	0
Minimum value:	[&:]	Minimum length:	4
Maximum value:	[zy:t][zy:t][Ost]	Maximum length:	52
Characters:	& 3 : @ A E I N O Q S U V Y Z [ ] a b d e f g h i j k l m n o p r s t u v w x y z ~			

PhonSylCLX Syll. phon. headword, CELEX charset

Type:	character	Null values:	0
Minimum value:	&:	Minimum length:	2
Maximum value:	zy:t-zy:t-Ost	Maximum length:	41
Characters:	& - 3 : @ A E I N O Q S U V Y Z a b d e f g h i j k l m n o p r s t u v w x y z ~		

PhonSylCPA Syll. phon. headword, CPA charset

Type:	character	Null values:	0
Minimum value:	@.gri:.m@nt	Minimum length:	2
Maximum value:	zy:t.zy:t.Ost	Maximum length:	41
Characters:	. / : @ A C E I J N O Q S T U Y Z ^ a b d e f g h i j k l m n o p q r s t u v w x y z ~		

PhonSylDISC Syll. phon. headword, DISC charset

Type:	character	Null values:	0
Minimum value:	$l-r6nd-SpOrt-l@r	Minimum length:	1
Maximum value:	|t-l&nt	Maximum length:	36
Characters:	# $ &) + - / 0 1 2 3 4 6 = @ A B E I J N O S U V W X Y Z ^ _ a b c d e f g h i j k l m n o p q r s t u v w x y z { | ~		

PhonSylSAM Syll. phon. headword, SAM-PA charset

Type:	character	Null values:	0
Minimum value:	/-f@nt-lIx	Minimum length:	2
Maximum value:	|:t-lant	Maximum length:	41
Characters:	- / 3 : @ A E I N O S U V Y Z a b d e f g h i j k l m n o p r s t u v w x y z { | ~		

PhonSylStBCLX Syll. phon. stem, CELEX charset (brackets)

Type:	character	Null values:	0
Minimum value:	[&:]	Minimum length:	4
Maximum value:	[zy:t][zy:t][Ost]	Maximum length:	52
Characters:	& 3 : @ A E I N O Q S U V Y Z [ ] a b d e f g h i j k l m n o p r s t u v w x y z ~		

PhonSylStCLX Syll. phon. stem, CELEX charset

Type:	character	Null values:	0
Minimum value:	&:	Minimum length:	2
Maximum value:	zy:t-zy:t-Ost	Maximum length:	41
Characters:	& - 3 : @ A E I N O Q S U V Y Z a b d e f g h i j k l m n o p r s t u v w x y z ~		

PhonSylStCPA Syll. phon. stem, CPA charset

Type:	character	Null values:	0
Minimum value:	@.gri:.m@nt	Minimum length:	2
Maximum value:	zy:t.zy:t.Ost	Maximum length:	41
Characters:	. / : @ A C E I J N O Q S T U Y Z ^ a b d e f g h i j k l m n o p q r s t u v w x y z ~		

PhonSylStDISC Syll. phon. stem, DISC charset

Type:	character	Null values:	0
Minimum value:	$l-r6nd-SpOrt-l@r	Minimum length:	1
Maximum value:	|t-l&nt	Maximum length:	36
Characters:	# $ &) + - / 0 1 2 3 4 6 = @ A B E I J N O S U V W X Y Z ^ _ a b c d e f g h i j k l m n o p q r s t u v w x y z { | ~		

PhonSylStSAM Syll. phon. stem, SAM-PA charset

Type:	character	Null values:	0
Minimum value:	/-f@nt-lIx	Minimum length:	2
Maximum value:	|:t-lant	Maximum length:	41
Characters:	- / 3 : @ A E I N O S U V Y Z a b d e f g h i j k l m n o p r s t u v w x y z { | ~		

PlurTant For nouns: plurale tantum

Type:	character	Null values:	0
Minimum value:	N	Minimum length:	1
Maximum value:	Y	Maximum length:	1
Characters:	N Y		

Prop For nouns: proper noun, labels

Type:	character	Null values:	51482
Minimum value:	B	Minimum length:	1
Maximum value:	P	Maximum length:	1
Characters:	B G P		

PropNum For nouns: proper noun, numeric

Type:	character	Null values:	51482
Minimum value:	1	Minimum length:	1
Maximum value:	3	Maximum length:	1
Characters:	1 2 3		

Sepa Separable␣

Type: character␣ Null values: 0␣
Minimum value: N␣ Minimum length: 1␣
Maximum value: Y␣ Maximum length: 1␣
Characters: N␣Y␣

SingTant For␣nouns:␣singulare␣tantum␣

Type: character␣ Null values: 0␣
Minimum value: N␣ Minimum length: 1␣
Maximum value: Y␣ Maximum length: 1␣
Characters: N␣Y␣

Stem Stem␣

Type: character␣ Null values: 0␣
Minimum value: A␣ Minimum length: 1␣
Maximum value: zytogen␣ Maximum length: 31␣
Characters: A␣B␣C␣D␣E␣F␣G␣H␣I␣J␣K␣L␣M␣N␣O␣P␣Q␣R␣S␣T␣U␣V␣W␣
X␣Y␣Z␣a␣b␣c␣d␣e␣f␣g␣h␣i␣j␣k␣l␣m␣n␣o␣p␣q␣r␣s␣t␣
u␣v␣w␣x␣y␣z␣

StemCnt Stem,␣number␣of␣letters␣

Type: numeric␣ Null values: 0␣
Minimum value: 1␣ Minimum length: 1␣
Maximum value: 31␣ Maximum length: 2␣
Characters: 0␣1␣2␣3␣4␣5␣6␣7␣8␣9␣

StemDia	Stem, diacritics

Type:	character	Null values:	0
Minimum value:	A	Minimum length:	1
Maximum value:	üppig	Maximum length:	31
Characters:	Ä Ö Ü ß ä é ö ü A B C D E F G H I J K L M N O P Q R S T U V W X Y Z a b c d e f g h i j k l m n o p q r s t u v w x y z		

StemRev	Stem, reversed

Type:	character	Null values:	0
Minimum value:	A	Minimum length:	1
Maximum value:	zzaJ	Maximum length:	31
Characters:	A B C D E F G H I J K L M N O P Q R S T U V W X Y Z a b c d e f g h i j k l m n o p q r s t u v w x y z		

StemSyl	Stem, syllabified

Type:	character	Null values:	0
Minimum value:	A	Minimum length:	1
Maximum value:	zy-to-gen	Maximum length:	40
Characters:	- = A B C D E F G H I J K L M N O P Q R S T U V W X Y Z a b c d e f g h i j k l m n o p q r s t u v w x y z		

StemSylChg	Spelling change, stem

Type:	character	Null values:	0
Minimum value:	N	Minimum length:	1
Maximum value:	Y	Maximum length:	1
Characters:	N Y		

StemSylCnt Stem, number of orthographic syllables

Type:	numeric	Null values:	0
Minimum value:	1	Minimum length:	1
Maximum value:	10	Maximum length:	2
Characters:	0 1 2 3 4 5 6 7 8 9		

StemSylDia Stem, syllabified, diacritics

Type:	character	Null values:	0
Minimum value:	A	Minimum length:	1
Maximum value:	üp-pig	Maximum length:	40
Characters:	Ä Ö Ü ß ä é ö ü - = A B C D E F G H I J K L M N O P Q R S T U V W X Y Z a b c d e f g h i j k l m n o p q r s t u v w x y z		

StrsPat Headword, stress pattern

Type:	character	Null values:	0
Minimum value:	00000001	Minimum length:	1110
Maximum value:	1	Maximum length:	10
Characters:	0 1		

Struc Structured segmentation

Type:	character	Null values:	0
Minimum value:	((((((alt),(er))), (tum)),(el)),(ei))	Minimum length:	3
Maximum value:	(zytogen)	Maximum length:	71
Characters:	( ) , A B C D E F G H I J K L M N O P Q R S T U V W X Y Z a b c d e f g h i j k l m n o p q r s t u v w x y z		

StrucAllo	Stem␣allomorphy,␣any␣level␣

Type:	character␣	Null values:	0␣
Minimum value:	N␣	Minimum length:	1␣
Maximum value:	Y␣	Maximum length:	1␣
Characters:	N␣Y␣		

StrucBrackLab	Structured␣segmentation,␣word␣class␣labels␣only␣

Type:	character␣	Null values:	0␣
Minimum value:	((␣)[n],(()[F])[N])[N]␣	Minimum length:	5␣
Maximum value:	()[V]␣	Maximum length:	115␣
Characters:	(␣)␣,␣.␣A␣B␣C␣D␣F␣I␣N␣O␣P␣Q␣R␣V␣[␣]␣c␣n␣p␣x␣|␣		

StrucLab	Structured␣segmentation,␣word␣class␣labels␣

Type:	character␣	Null values:	0␣
Minimum value:	((((((alt)[A],(er)␣ [V|A.])[V])[N],␣ (tum)[N|N.])[N],␣ (el)[V|N.])[V],␣ (ei)[N|V.])[N]␣	Minimum length:	6␣
Maximum value:	(zytogen)[A]␣	Maximum length:	139␣
Characters:	(␣)␣,␣.␣A␣B␣C␣D␣E␣F␣G␣H␣I␣J␣K␣L␣M␣N␣O␣P␣Q␣R␣S␣ T␣U␣V␣W␣X␣Y␣Z␣[␣]␣a␣b␣c␣d␣e␣f␣g␣h␣i␣j␣k␣l␣m␣n␣ o␣p␣q␣r␣s␣t␣u␣v␣w␣x␣y␣z␣|␣		

StrucOpac	Opacity,␣any␣

Type:	character␣	Null values:	0␣
Minimum value:	N␣	Minimum length:	1␣
Maximum value:	Y␣	Maximum length:	1␣
Characters:	N␣Y␣		

StrucUml Umlaut, any level

Type:	character	Null values:	0
Minimum value:	N	Minimum length:	1
Maximum value:	Y	Maximum length:	1
Characters:	N Y		

StStrsPat Stem, stress pattern

Type:	character	Null values:	0
Minimum value:	0	Minimum length:	1
Maximum value:	1110	Maximum length:	10
Characters:	0 1		

StSylCnt Stem, number of phonetic syllables

Type:	numeric	Null values:	0
Minimum value:	1	Minimum length:	1
Maximum value:	10	Maximum length:	2
Characters:	0 1 2 3 4 5 6 7 8 9		

SubClassP For pronouns: subclasses, labels

Type:	character	Null values:	51612
Minimum value:	demonstrative	Minimum length:	8
Maximum value:	relative	Maximum length:	13
Characters:	a c d e f g i l m n o p r s t v x		

SubClassPNum For pronouns: subclasses, numeric

Type:	character	Null values:	51612
Minimum value:	1	Minimum length:	1
Maximum value:	8	Maximum length:	1
Characters:	1 2 3 4 5 6 7 8		

SubClassV	For␣verbs:␣subclasses,␣labels␣

Type:	character␣	Null values:	0␣
Minimum value:	ac␣	Minimum length:	1␣
Maximum value:	r␣	Maximum length:	3␣
Characters:	a␣c␣i␣l␣m␣r␣		

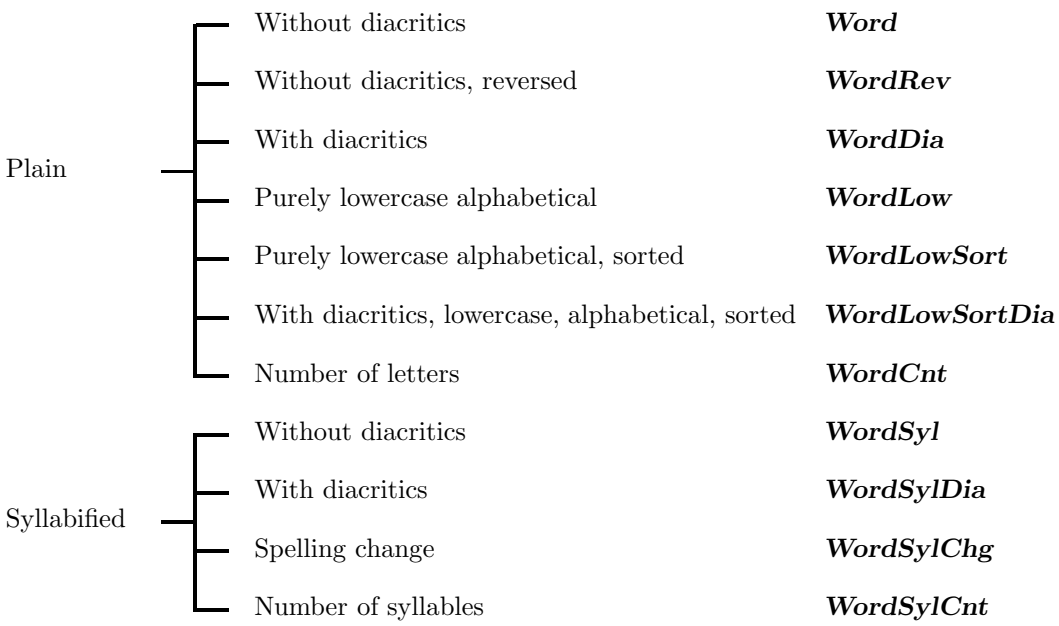
SubClassVNum	For␣verbs:␣subclasses,␣numeric␣

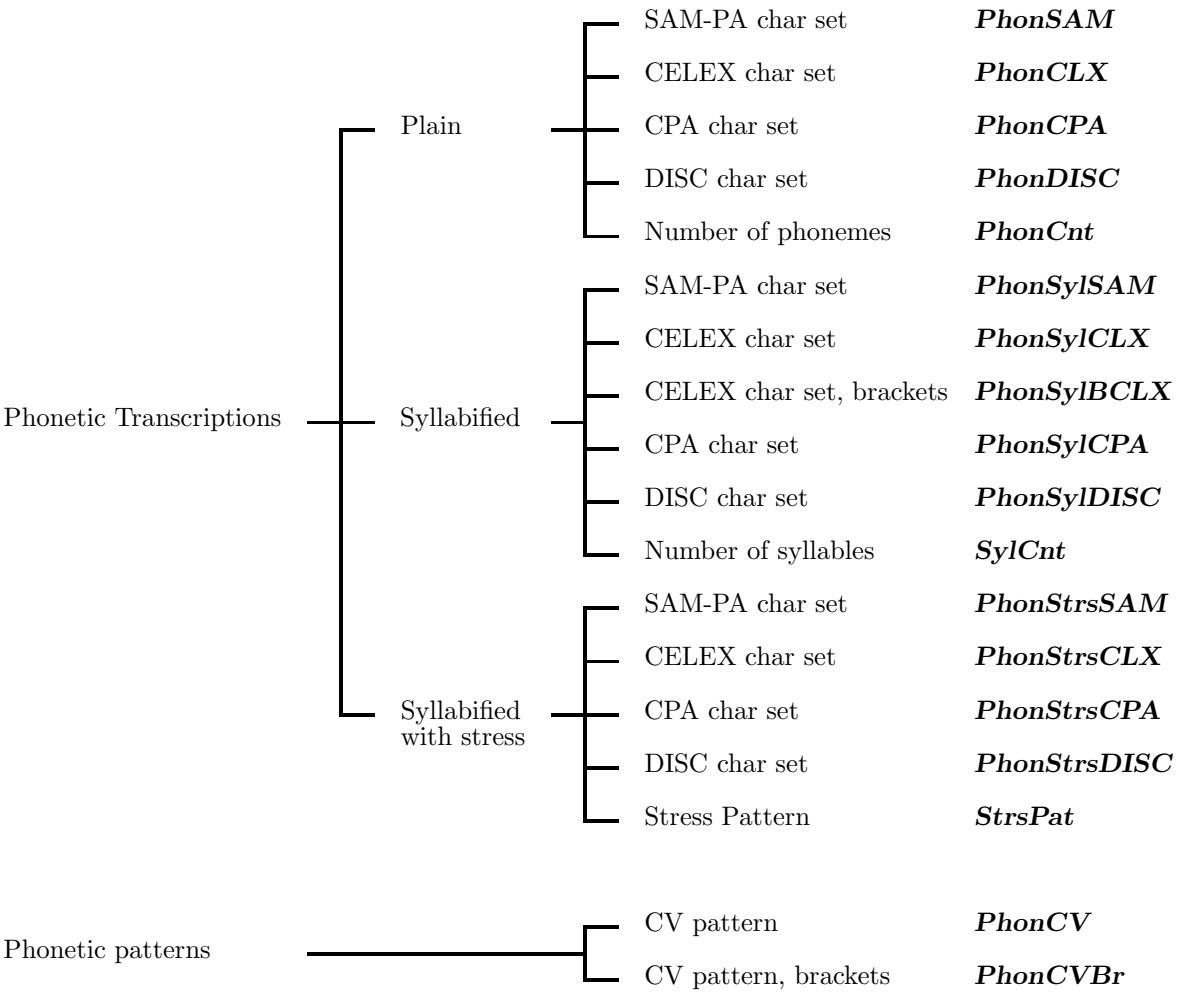
Type:	character␣	Null values:	0␣
Minimum value:	12␣	Minimum length:	1␣
Maximum value:	6␣	Maximum length:	3␣
Characters:	1␣2␣3␣4␣5␣6␣		

SylCnt	Headword,␣number␣of␣phonetic␣syllables␣

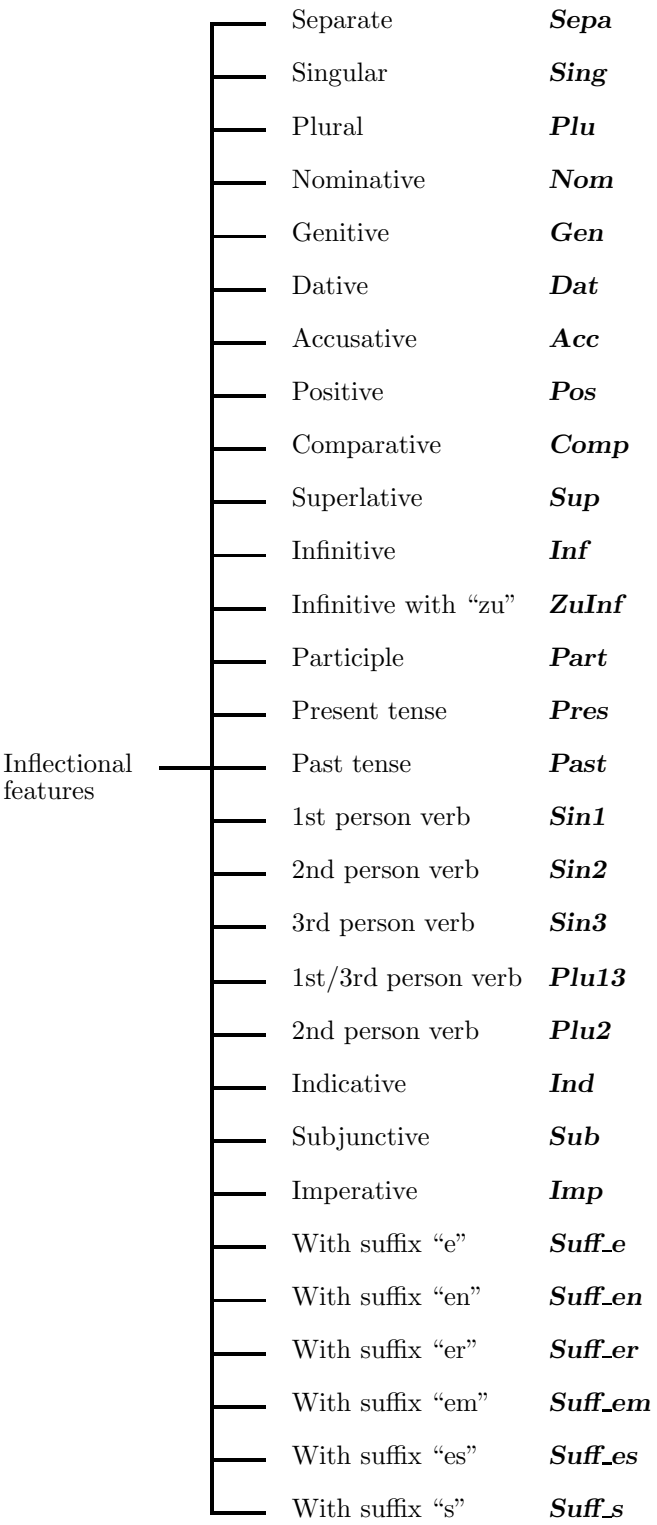
Type:	numeric␣	Null values:	0␣
Minimum value:	1␣	Minimum length:	1␣
Maximum value:	10␣	Maximum length:	2␣
Characters:	0␣1␣2␣3␣4␣5␣6␣7␣8␣9␣		

8 ORTHOGRAPHY OF GERMAN WORDFORMS (D25)

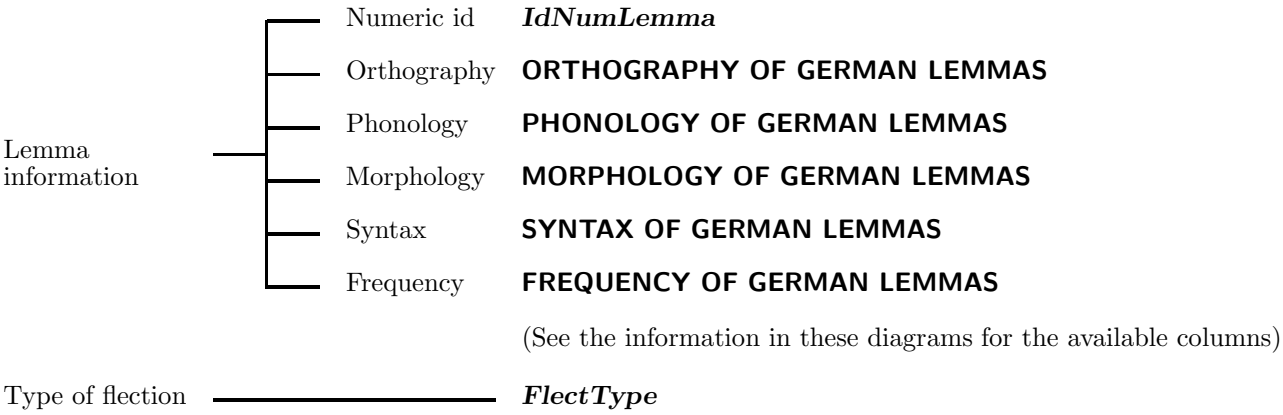



Phonetic Transcriptions
Plain
SAM-PA char set
PhonSAM
CELEX char set
PhonCLX
CPA char set
PhonCPA
DISC char set
PhonDISC
Number of phonemes
PhonCnt
Syllabified
SAM-PA char set
PhonSylSAM
CELEX char set
PhonSylCLX
CELEX char set, brackets
PhonSylBCLX
CPA char set
PhonSylCPA
DISC char set
PhonSylDISC
Number of syllables
SylCnt
Syllabified with stress
SAM-PA char set
PhonStrsSAM
CELEX char set
PhonStrsCLX
CPA char set
PhonStrsCPA
DISC char set
PhonStrsDISC
Stress Pattern
StrsPat
Phonetic patterns
CV pattern
PhonCV
CV pattern, brackets
PhonCVBr

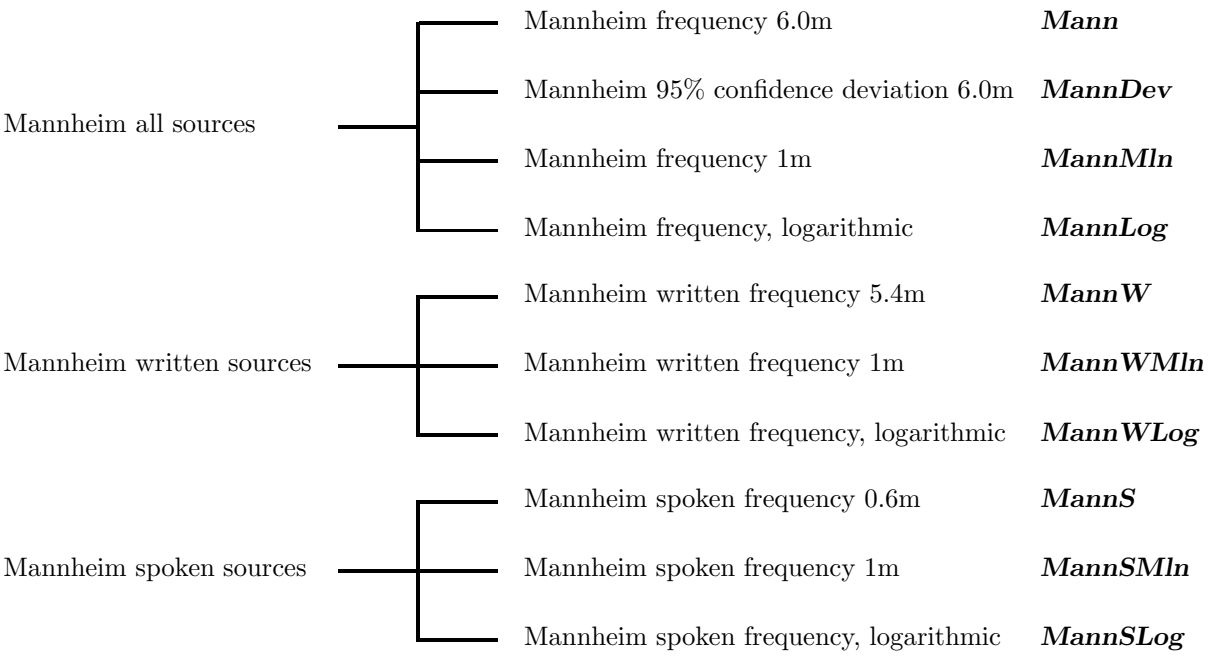
10 MORPHOLOGY OF GERMAN WORDFORMS (D25)



11 MORPHOLOGY OF GERMAN WORDFORMS (D25)



12 FREQUENCY OF GERMAN WORDFORMS (D25)



Acc	Inflectional␣feature:␣accusative␣

Type:	character␣	Null values:	0␣
Minimum value:	N␣	Minimum length:	1␣
Maximum value:	Y␣	Maximum length:	1␣
Characters:	N␣Y␣		

Comp	Inflectional␣feature:␣comparative␣

Type:	character␣	Null values:	0␣
Minimum value:	N␣	Minimum length:	1␣
Maximum value:	Y␣	Maximum length:	1␣
Characters:	N␣Y␣		

Dat	Inflectional␣feature:␣dative␣

Type:	character␣	Null values:	0␣
Minimum value:	N␣	Minimum length:	1␣
Maximum value:	Y␣	Maximum length:	1␣
Characters:	N␣Y␣		

FlectType	Type␣of␣flection␣

Type:	character␣	Null values:	0␣
Minimum value:	0␣	Minimum length:	1␣
Maximum value:	z/␣	Maximum length:	23␣
Characters:	,␣/␣0␣1␣2␣3␣4␣5␣6␣7␣8␣9␣A␣E␣I␣K␣P␣S␣X␣a␣c␣d␣g␣ i␣m␣n␣o␣p␣r␣s␣u␣w␣z␣		

Gen	Inflectional␣feature:␣genitive␣

Type:	character␣	Null values:	0␣
Minimum value:	N␣	Minimum length:	1␣
Maximum value:	Y␣	Maximum length:	1␣
Characters:	N␣Y␣		

IdNum	Word␣number␣

Type:	numeric␣	Null values:	0␣
Minimum value:	1␣	Minimum length:	1␣
Maximum value:	365530␣	Maximum length:	6␣
Characters:	0␣1␣2␣3␣4␣5␣6␣7␣8␣9␣		

Imp	Inflectional␣feature:␣imperative␣

Type:	character␣	Null values:	0␣
Minimum value:	N␣	Minimum length:	1␣
Maximum value:	Y␣	Maximum length:	1␣
Characters:	N␣Y␣		

Ind	Inflectional␣feature:␣indicative␣

Type:	character␣	Null values:	0␣
Minimum value:	N␣	Minimum length:	1␣
Maximum value:	Y␣	Maximum length:	1␣
Characters:	N␣Y␣		

Inf	Inflectional␣feature:␣infinitive␣

Type:	character␣	Null values:	0␣
Minimum value:	N␣	Minimum length:	1␣
Maximum value:	Y␣	Maximum length:	1␣
Characters:	N␣Y␣		

Mann	`Mannheim␣frequency␣`

Type:	`numeric␣`	Null values:	`0␣`
Minimum value:	`0␣`	Minimum length:	`1␣`
Maximum value:	`150507␣`	Maximum length:	`6␣`
Characters:	`0 1 2 3 4 5 6 7 8 9␣`		

MannDev	`Mannheim␣frequency␣deviation␣`

Type:	`numeric␣`	Null values:	`0␣`
Minimum value:	`0␣`	Minimum length:	`1␣`
Maximum value:	`946884␣`	Maximum length:	`6␣`
Characters:	`0 1 2 3 4 5 6 7 8 9␣`		

MannLog	`Mannheim␣frequency,␣logarithmic␣`

Type:	`numeric␣`	Null values:	`0␣`
Minimum value:	`0␣`	Minimum length:	`1␣`
Maximum value:	`4.4029␣`	Maximum length:	`6␣`
Characters:	`. 0 1 2 3 4 5 6 7 8 9␣`		

MannMln	`Mannheim␣frequency␣(1,000,000)␣`

Type:	`numeric␣`	Null values:	`0␣`
Minimum value:	`0␣`	Minimum length:	`1␣`
Maximum value:	`25287␣`	Maximum length:	`5␣`
Characters:	`0 1 2 3 4 5 6 7 8 9␣`		

MannS	`Mannheim␣spoken␣frequency␣0.6m␣`

Type:	`numeric␣`	Null values:	`0␣`
Minimum value:	`0␣`	Minimum length:	`1␣`
Maximum value:	`15565␣`	Maximum length:	`5␣`
Characters:	`0 1 2 3 4 5 6 7 8 9␣`		

MannSLog	`Mannheim␣spoken␣frequency,␣logarithmic␣`

Type:	`numeric␣`	Null values:	`0␣`
Minimum value:	`0␣`	Minimum length:	`1␣`
Maximum value:	`4.4234␣`	Maximum length:	`6␣`
Characters:	`. 0 1 2 3 4 5 6 7 8 9␣`		

MannSMln	`Mannheim␣spoken␣frequency␣(1,000,000)␣`

Type:	`numeric␣`	Null values:	`0␣`
Minimum value:	`0␣`	Minimum length:	`1␣`
Maximum value:	`26508␣`	Maximum length:	`5␣`
Characters:	`0 1 2 3 4 5 6 7 8 9␣`		

MannW	`Mannheim␣written␣frequency␣5.4m␣`

Type:	`numeric␣`	Null values:	`0␣`
Minimum value:	`0␣`	Minimum length:	`1␣`
Maximum value:	`134942␣`	Maximum length:	`6␣`
Characters:	`0 1 2 3 4 5 6 7 8 9␣`		

MannWLog	`Mannheim␣written␣frequency,␣logarithmic␣`

Type:	`numeric␣`	Null values:	`0␣`
Minimum value:	`0␣`	Minimum length:	`1␣`
Maximum value:	`4.4006␣`	Maximum length:	`6␣`
Characters:	`. 0 1 2 3 4 5 6 7 8 9␣`		

MannWMln	`Mannheim␣written␣frequency␣(1,000,000)␣`

Type:	`numeric␣`	Null values:	`0␣`
Minimum value:	`0␣`	Minimum length:	`1␣`
Maximum value:	`25153␣`	Maximum length:	`5␣`
Characters:	`0 1 2 3 4 5 6 7 8 9␣`		

Nom	Inflectional␣feature:␣nominative␣

Type:	character␣	Null values:	0␣
Minimum value:	N␣	Minimum length:	1␣
Maximum value:	Y␣	Maximum length:	1␣
Characters:	N␣Y␣		

Part	Inflectional␣feature:␣participle␣

Type:	character␣	Null values:	0␣
Minimum value:	N␣	Minimum length:	1␣
Maximum value:	Y␣	Maximum length:	1␣
Characters:	N␣Y␣		

Past	Inflectional␣feature:␣past␣tense␣

Type:	character␣	Null values:	0␣
Minimum value:	N␣	Minimum length:	1␣
Maximum value:	Y␣	Maximum length:	1␣
Characters:	N␣Y␣		

PhonCLX	Phon.␣wordform,␣CELEX␣charset␣

Type:	character␣	Null values:	0␣
Minimum value:	&:.␣	Minimum length:	3␣
Maximum value:	z.y:.ts.␣	Maximum length:	61␣
Characters:	&␣.␣3␣:␣@␣A␣E␣I␣N␣O␣Q␣S␣U␣V␣Y␣Z␣a␣b␣d␣e␣f␣g␣ h␣i␣j␣k␣l␣m␣n␣o␣p␣r␣s␣t␣u␣v␣w␣x␣y␣z␣˜␣		

PhonCnt Word, number of phonemes

Type:	numeric	Null values:	0
Minimum value:	1	Minimum length:	1
Maximum value:	29	Maximum length:	2
Characters:	0 1 2 3 4 5 6 7 8 9		

PhonCPA Phon. wordform, CPA charset

Type:	character	Null values:	0
Minimum value:	@.d.v.A:.n.t.I.J/.	Minimum length:	3
Maximum value:	z.y:.t.z.y:.t.O.s.t.	Maximum length:	61
Characters:	. / : @ A C E I J N O Q S T U Y Z ^ a b d e f g h i j k l m n o p q r s t u v w x y z ~		

PhonCV Wordform, phon. CV pattern

Type:	character	Null values:	0
Minimum value:	CCCVC	Minimum length:	2
Maximum value:	VVCCC-VVC-CVV-CVC	Maximum length:	43
Characters:	- C V		

PhonCVBr Wordform, phon. CV pattern, with brackets

Type:	character	Null values:	0
Minimum value:	[CCCVCCCC]	Minimum length:	4
Maximum value:	[V][VV][CV[C]V]	Maximum length:	54
Characters:	C V [ ]		

PhonDISC Phon. wordform, DISC charset

Type: character — Null values: 0
Minimum value: $lr6ndSpOrtl@r — Minimum length: 1
Maximum value: |z@n — Maximum length: 29
Characters: # $ &) + / 0 1 2 3 4 6 = @ A B E I J N O S U V W X Y Z ^ _ a b c d e f g h i j k l m n o p q r s t u v w x y z { | ~

PhonSAM Phon. wordform, SAM-PA charset

Type: character — Null values: 0
Minimum value: /.f.@.n.t.l.I.x. — Minimum length: 3
Maximum value: |:.z.@.n. — Maximum length: 61
Characters: . / 3 : @ A E I N O S U V Y Z a b d e f g h i j k l m n o p r s t u v w x y z { | ~

PhonStrsCLX Syll. phon. wordform, with stress, CELEX charset

Type: character — Null values: 0
Minimum value: &:-d@ 'an — Minimum length: 3
Maximum value: zy:t-zy:t-'Ost — Maximum length: 45
Characters: " & ' - 3 : @ A E I N O Q S U V Y Z a b d e f g h i j k l m n o p r s t u v w x y z ~

PhonStrsCPA Syll. phon. wordform, with stress, CPA charset

Type: character — Null values: 0
Minimum value: 'A/ — Minimum length: 3
Maximum value: zy:t.zy:t.'Ost — Maximum length: 45
Characters: " ' . / : @ A C E I J N O Q S T U Y Z ^ a b d e f g h i j k l m n o p q r s t u v w x y z ~

PhonStrsDISC

Syll. phon. wordform, with stress, DISC charset

Type:	character	Null values:	0
Minimum value:	&-'=a-li-@	Minimum length:	2
Maximum value:	|lt 'Wn	Maximum length:	40

Characters: " # $ & ') + - / 0 1 2 3 4 6 = @ A B E I J N O S U V W X Y Z ^ _ a b c d e f g h i j k l m n o p q r s t u v w x y z { | ~

PhonStrsSAM

Syll. phon. wordform, with stress, SAM-PA charset

Type:	character	Null values:	0
Minimum value:	"/-f@nt-lI-x@	Minimum length:	3
Maximum value:	|:lt "ain	Maximum length:	45

Characters: " % - / 3 : @ A E I N O S U V Y Z a b d e f g h i j k l m n o p r s t u v w x y z { | ~

PhonSylBCLX

Syll. phon. wordform, CELEX charset (brackets)

Type:	character	Null values:	0
Minimum value:	[&:]	Minimum length:	4
Maximum value:	[zy:ts]	Maximum length:	56

Characters: & 3 : @ A E I N O Q S U V Y Z [ ] a b d e f g h i j k l m n o p r s t u v w x y z ~

PhonSylCLX

Syll. phon. wordform, CELEX charset

Type:	character	Null values:	0
Minimum value:	&:	Minimum length:	2
Maximum value:	zy:ts	Maximum length:	44

Characters: & - 3 : @ A E I N O Q S U V Y Z a b d e f g h i j k l m n o p r s t u v w x y z ~

PhonSylCPA Syll. phon. wordform, CPA charset

Type:	character	Null values:	0
Minimum value:	@.gri:.m@nC/	Minimum length:	2
Maximum value:	zy:t.zy:t.Ost	Maximum length:	44
Characters:	. / : @ A C E I J N O Q S T U Y Z ^ a b d e f g h i j k l m n o p q r s t u v w x y z ~		

PhonSylDISC Syll. phon. wordform, DISC charset

Type:	character	Null values:	0
Minimum value:	$l-r6nd-SpOrt-l@r	Minimum length:	1
Maximum value:	|t-st@s	Maximum length:	39
Characters:	# $ &) + - / 0 1 2 3 4 6 = @ A B E I J N O S U V W X Y Z ^ _ a b c d e f g h i j k l m n o p q r s t u v w x y z { | ~		

PhonSylSAM Syll. phon. wordform, SAM-PA charset

Type:	character	Null values:	0
Minimum value:	/-f@nt-lI-x@	Minimum length:	2
Maximum value:	|:tst	Maximum length:	44
Characters:	- / 3 : @ A E I N O S U V Y Z a b d e f g h i j k l m n o p r s t u v w x y z { | ~		

Plu Inflectional feature: plural

Type:	character	Null values:	0
Minimum value:	N	Minimum length:	1
Maximum value:	Y	Maximum length:	1
Characters:	N Y		

Plu13	Inflectional feature: 1st/3rd person plural verb

Type:	character	Null values:	0
Minimum value:	N	Minimum length:	1
Maximum value:	Y	Maximum length:	1
Characters:	N Y		

Plu2	Inflectional feature: 2nd person plural verb

Type:	character	Null values:	0
Minimum value:	N	Minimum length:	1
Maximum value:	Y	Maximum length:	1
Characters:	N Y		

Pos	Inflectional feature: positive

Type:	character	Null values:	0
Minimum value:	N	Minimum length:	1
Maximum value:	Y	Maximum length:	1
Characters:	N Y		

Pres	Inflectional feature: present tense

Type:	character	Null values:	0
Minimum value:	N	Minimum length:	1
Maximum value:	Y	Maximum length:	1
Characters:	N Y		

Sepa	Separated wordform

Type:	character	Null values:	0
Minimum value:	N	Minimum length:	1
Maximum value:	Y	Maximum length:	1
Characters:	N Y		

Sin	Inflectional␣feature:␣singular␣

Type:	character␣	Null values:	0␣
Minimum value:	N␣	Minimum length:	1␣
Maximum value:	Y␣	Maximum length:	1␣
Characters:	N␣Y␣		

Sin1	Inflectional␣feature:␣1st␣person␣singular␣verb␣

Type:	character␣	Null values:	0␣
Minimum value:	N␣	Minimum length:	1␣
Maximum value:	Y␣	Maximum length:	1␣
Characters:	N␣Y␣		

Sin2	Inflectional␣feature:␣2nd␣person␣singular␣verb␣

Type:	character␣	Null values:	0␣
Minimum value:	N␣	Minimum length:	1␣
Maximum value:	Y␣	Maximum length:	1␣
Characters:	N␣Y␣		

Sin3	Inflectional␣feature:␣3rd␣person␣singular␣verb␣

Type:	character␣	Null values:	0␣
Minimum value:	N␣	Minimum length:	1␣
Maximum value:	Y␣	Maximum length:	1␣
Characters:	N␣Y␣		

StrsPat	Word,␣stress␣pattern␣

Type:	character␣	Null values:	0␣
Minimum value:	0␣00␣1␣	Minimum length:	1␣
Maximum value:	11100␣	Maximum length:	12␣
Characters:	0␣1␣2␣		

Sub	Inflectional feature: subjunctive

Type:	character	Null values:	0
Minimum value:	N	Minimum length:	1
Maximum value:	Y	Maximum length:	1
Characters:	N Y		

Suff_e	Inflectional feature: with suffix "e"

Type:	character	Null values:	0
Minimum value:	N	Minimum length:	1
Maximum value:	Y	Maximum length:	1
Characters:	N Y		

Suff_em	Inflectional feature: with suffix "em"

Type:	character	Null values:	0
Minimum value:	N	Minimum length:	1
Maximum value:	Y	Maximum length:	1
Characters:	N Y		

Suff_en	Inflectional feature: with suffix "en"

Type:	character	Null values:	0
Minimum value:	N	Minimum length:	1
Maximum value:	Y	Maximum length:	1
Characters:	N Y		

Suff_er	Inflectional feature: with suffix "er"

Type:	character	Null values:	0
Minimum value:	N	Minimum length:	1
Maximum value:	Y	Maximum length:	1
Characters:	N Y		

Suff_es	Inflectional feature: with suffix "es"

Type:	character	Null values:	0
Minimum value:	N	Minimum length:	1
Maximum value:	Y	Maximum length:	1
Characters:	N Y		

Suff_s	Inflectional feature: with suffix "s"

Type:	character	Null values:	0
Minimum value:	N	Minimum length:	1
Maximum value:	Y	Maximum length:	1
Characters:	N Y		

Sup	Inflectional feature: superlative

Type:	character	Null values:	0
Minimum value:	N	Minimum length:	1
Maximum value:	Y	Maximum length:	1
Characters:	N Y		

SylCnt	Word, number of phonetic syllables

Type:	numeric	Null values:	0
Minimum value:	1	Minimum length:	1
Maximum value:	11	Maximum length:	2
Characters:	0 1 2 3 4 5 6 7 8 9		

Word	Word␣

Type:	character␣	Null values:	0␣
Minimum value:	A␣	Minimum length:	1␣
Maximum value:	zytogenes␣	Maximum length:	33␣
Characters:	A␣B␣C␣D␣E␣F␣G␣H␣I␣J␣K␣L␣M␣N␣O␣P␣Q␣R␣S␣T␣U␣V␣ W␣X␣Y␣Z␣a␣b␣c␣d␣e␣f␣g␣h␣i␣j␣k␣l␣m␣n␣o␣p␣q␣r␣s␣ t␣u␣v␣w␣x␣y␣z␣		

WordCnt	Word,␣number␣of␣letters␣

Type:	numeric␣	Null values:	0␣
Minimum value:	1␣	Minimum length:	1␣
Maximum value:	33␣	Maximum length:	2␣
Characters:	0␣1␣2␣3␣4␣5␣6␣7␣8␣9␣		

WordDia	Word,␣diacritics␣

Type:	character␣	Null values:	0␣
Minimum value:	A␣	Minimum length:	1␣
Maximum value:	üppigstes␣	Maximum length:	33␣
Characters:	A␣B␣C␣D␣E␣F␣G␣H␣I␣J␣K␣L␣M␣N␣O␣P␣Q␣R␣S␣T␣U␣V␣ W␣X␣Y␣Z␣a␣b␣c␣d␣e␣f␣g␣h␣i␣j␣k␣l␣m␣n␣o␣p␣q␣r␣s␣ t␣u␣v␣w␣x␣y␣z␣Ä␣Ö␣Ü␣ß␣ä␣é␣ö␣ü␣		

WordLow	Word,␣lowercase,␣alphabetical␣

Type:	character␣	Null values:	0␣
Minimum value:	a␣	Minimum length:	1␣
Maximum value:	zytostoms␣	Maximum length:	33␣
Characters:	a␣b␣c␣d␣e␣f␣g␣h␣i␣j␣k␣l␣m␣n␣o␣p␣q␣r␣s␣t␣u␣v␣ w␣x␣y␣z␣		

WordLowSort	Word, lowercase, alphabetical, sorted

Type:	character	Null values:	0
Minimum value:	a	Minimum length:	1
Maximum value:	z	Maximum length:	33
Characters:	a b c d e f g h i j k l m n o p q r s t u v w x y z		

WordLowSortDia	Word, lowercase, sorted, diacritics

Type:	character	Null values:	0
Minimum value:	a	Minimum length:	1
Maximum value:	ü	Maximum length:	33
Characters:	a b c d e f g h i j k l m n o p q r s t u v w x y z ß ä é ö ü		

WordRev	Word, reversed

Type:	character	Null values:	0
Minimum value:	A	Minimum length:	1
Maximum value:	zzaJ	Maximum length:	33
Characters:	A B C D E F G H I J K L M N O P Q R S T U V W X Y Z a b c d e f g h i j k l m n o p q r s t u v w x y z		

WordSyl	Word, syllabified

Type:	character	Null values:	0
Minimum value:	A	Minimum length:	1
Maximum value:	zy-to-gen	Maximum length:	43
Characters:	- = A B C D E F G H I J K L M N O P Q R S T U V W X Y Z a b c d e f g h i j k l m n o p q r s t u v w x y z		

WordSylChg Spelling␣change,␣word␣

Type:	character␣	Null values:	0␣
Minimum value:	N␣	Minimum length:	1␣
Maximum value:	Y␣	Maximum length:	1␣
Characters:	N␣Y␣		

WordSylCnt Word,␣number␣of␣orthographic␣syllables␣

Type:	numeric␣	Null values:	0␣
Minimum value:	1␣	Minimum length:	1␣
Maximum value:	11␣	Maximum length:	2␣
Characters:	0␣1␣2␣3␣4␣5␣6␣7␣8␣9␣		

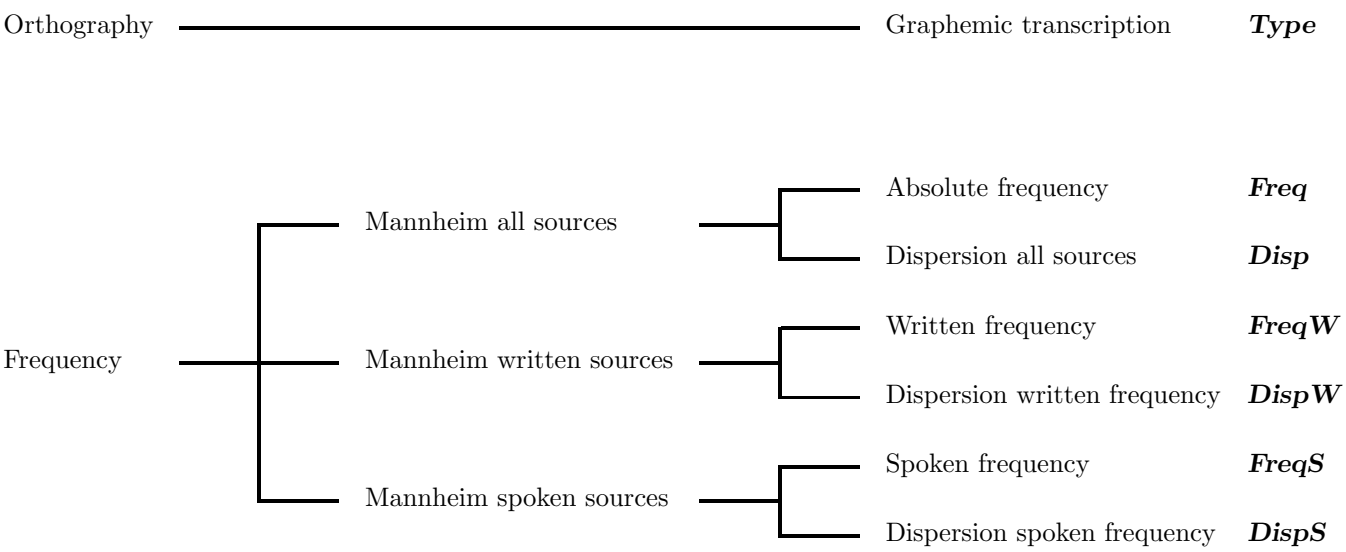
WordSylDia Word,␣syllabified,␣diacritics␣

Type:	character␣	Null values:	0␣
Minimum value:	A␣	Minimum length:	1␣
Maximum value:	üp-pigst␣	Maximum length:	43␣
Characters:	- =␣A␣B␣C␣D␣E␣F␣G␣H␣I␣J␣K␣L␣M␣N␣O␣P␣Q␣R␣S␣T␣ U␣V␣W␣X␣Y␣Z␣a␣b␣c␣d␣e␣f␣g␣h␣i␣j␣k␣l␣m␣n␣o␣p␣q␣ r␣s␣t␣u␣v␣w␣x␣y␣z␣Ä␣Ö␣Ü␣ß␣ä␣é␣ö␣ü␣		

ZuInf Inflectional␣feature:␣infinitive␣with␣"zu"␣

Type:	character␣	Null values:	0␣
Minimum value:	N␣	Minimum length:	1␣
Maximum value:	Y␣	Maximum length:	1␣
Characters:	N␣Y␣		

13 GERMAN MANNHEIM CORPUS TYPES (D25)



Disp	`Dispersion␣`

Type:	`numeric␣`	Null values:	`0␣`
Minimum value:	`0␣`	Minimum length:	`1␣`
Maximum value:	`316␣`	Maximum length:	`3␣`
Characters:	`0␣1␣2␣3␣4␣5␣6␣7␣8␣9␣`		

DispS	`Dispersion␣spoken␣sources␣`

Type:	`numeric␣`	Null values:	`0␣`
Minimum value:	`0␣`	Minimum length:	`1␣`
Maximum value:	`224␣`	Maximum length:	`3␣`
Characters:	`0␣1␣2␣3␣4␣5␣6␣7␣8␣9␣`		

DispW	`Dispersion␣written␣sources␣`

Type:	`numeric␣`	Null values:	`0␣`
Minimum value:	`0␣`	Minimum length:	`1␣`
Maximum value:	`92␣`	Maximum length:	`2␣`
Characters:	`0␣1␣2␣3␣4␣5␣6␣7␣8␣9␣`		

Freq	`Absolute␣frequency␣`

Type:	`numeric␣`	Null values:	`0␣`
Minimum value:	`1␣`	Minimum length:	`1␣`
Maximum value:	`218826␣`	Maximum length:	`6␣`
Characters:	`0␣1␣2␣3␣4␣5␣6␣7␣8␣9␣`		

FreqS	`Spoken␣frequency,␣0.6m␣`

Type:	`numeric␣`	Null values:	`0␣`
Minimum value:	`0␣`	Minimum length:	`1␣`
Maximum value:	`17888␣`	Maximum length:	`5␣`
Characters:	`0␣1␣2␣3␣4␣5␣6␣7␣8␣9␣`		

FreqW	Written frequency, 5.4m

Type:	numeric	Null values:	0
Minimum value:	0	Minimum length:	1
Maximum value:	203894	Maximum length:	6
Characters:	0 1 2 3 4 5 6 7 8 9		

Type	Graphemic transcription

Type:	character	Null values:	0
Minimum value:	A’dam	Minimum length:	1
Maximum value:	üssel	Maximum length:	92
Characters:	Ä Ö Ü ß ä ö ü ! " ’ ( ) * + , - . / 0 1 2 3 4 5 6 7 8 9 : ; = @ A B C D E F G H I J K L M N O P Q R S T U V W X Y Z a b c d e f g h i j k l m n o p q r s t u v w x y z		

	Infiniv	Indicativ Präsens	Indicativ Präteritum	Konjunktiv Präteritum	Imperativ	Partizip des Perfekts
101	backen	backe, bäckst, bäckt	buk, ∼ (e)st, backte	büke	back(e)	gebacken
102	befehlen	befehle, befiehlst, befiehlt	befahl	beföhle(befähle)	befiehl	befohlen
103	befleißen	befleiß/e, ∼ (es)t, ∼ t	befliß, beflissest	beflisse	befleiß(e)	beflissen
104	beginnen	beginn/e, ∼ st, ∼ t	begann	begönne (begänne)	beginn(e)	begonnen
105	beißen	beiß/e, ∼ (es)t, ∼ t	biß, bissest	bisse	beiß(e)	gebissen
106	bergen	berge, birgst, birgt	barg	bürge (bärge)	birg	geborgen
107	bersten	berste, birst (berstest), birst (berstet)	barst (borst, berstete), ∼ est	börste (bärste)	birst	geborsten
108	bewegen	beweg/e, ∼ st, ∼ t	bewegte (bewog)	bewöge	beweg(e)	bewegt bewogen
109	biegen	bieg/e, ∼ st, ∼ t	bog	böge	bieg(e)	gebogen
110	bieten	biet/e, ∼ (e)st, ∼ et	bot, ∼ (e)st	böte	biet(e)	geboten
111	binden	bind/e, ∼ est, ∼ et	band, ∼ (e)st	bände	bind(e)	gebunden
112	bitten	bitt/e, ∼ est, ∼ et	bat, ∼ (e)st	bäte	bitte	gebeten
113	blasen	blase, bläs(es)t,bläst	blies, ∼ est	bliese	blas(e)	geblasen
114	bleiben	bleib/e, ∼ st, ∼ t	blieb, ∼ (e)st	bliebe	bleib(e)	geblieben
115	braten	brate, brätst, brät	briet, ∼ (e)st	briete	brat(e)	gebraten
116	brechen	breche, brichst, bricht	brach	bräche	brich	gebrochen
117	brennen	brenn/e, ∼ st, ∼ t	brannte	brennte	brenne	gebrannt
118	bringen	bring/e, ∼ st, ∼ t	brachte	brächte	bring(e)	gebracht
119	denken	denk/e, ∼ st, ∼ t	dachte	dächte	denk(e)	gedacht
120	dingen	ding/e, ∼ st, ∼ t	dang (dingte)	ding(e)te, (dünge, dänge)	ding(e)	gedungen (gedingt)
121	dreschen	dresche, drisch(e)st, drischt	drosch (drasch), ∼ (e)st	drösche	drisch	gedroschen
122	dringen	dring/e, ∼ st, ∼ t	drang, ∼ (e)st	dränge	dring(e)	gedrungen
123	dünken	dünkt (deucht)	dünkte (deuchte)	—	—	gedünkt
124	dürfen	darf, ∼ st, ∼ , dürfen	durfte	dürfte	—	gedurft
125	empfehlen	empfehle,∼ fiehlst, ∼ fiehlt	empfahl	empföhle	empfiehl	empfohlen
126	erbleichen	erbleich/e, ∼ st, ∼ t	erbleichte (erblich)	erbleichte (erbliche)	erbleich(e)	erbleicht (erblichen)
127	erkiesen	erkies/e, ∼ (es)t, ∼ t	erkor	erköre	erkies(e)	erkoren
128	erlöschen	erlösche, erlisch(e)st, erlischt	erlosch, ∼ est	erlösche	erlisch	erloschen
129	essen	esse, issest (ißt), ißt	aß, ∼ est	äße	iß	gegessen
130	fahren	fahre, fährst, fährt	fuhr, ∼ (e)st	führe	fahr(e)	gefahren

Table of conjugations of German verbs

	Infiniv	Indicativ Präsens	Indicativ Präteritum	Konjunktiv Präteritum	Imperativ	Partizip des Perfekts
131	fallen	falle, fällst, fällt	fiel	fiele	fall(e)	gefallen
132	fangen	fange, fängst, fängt	fing	finge	fang(e)	gefangen
133	fechten	fechte, fichst, ficht	focht, ∼ (e)st	föchte	ficht	gefochten
134	finden	find/e, ∼ est, ∼ et	fand, ∼ (e)st	fände	find(e)	gefunden
135	flechten	flechte, flichst, flicht	flocht, ∼ (e)st	flöchte	flicht	geflochten
136	fliegen	flieg/e, ∼ st, ∼ t	flog, ∼ (e)st	flöge	flieg(e)	geflogen
137	fliehen	flieh/e, ∼ st, ∼ t	floh, ∼ (e)st	flöhe	flieh(e)	geflohen
138	fließen	fließ/e, ∼ (es)t, ∼ t	floß, flossest	flösse	fließ(e)	geflossen
139	fressen	fresse, frissest (frißt), frißt	fraß, ∼ est	fräße	friß	gefressen
140	frieren	frier/e, ∼ st, ∼ t	fror	fröre	frier(e)	gefroren
141	gären	gär/e, ∼ st, ∼ t	gor (gärte)	göre (gärte)	gär(e)	gegoren (gegärt)
142	gebären	gebäre, gebierst, gebiert gebärst, gebärt	gebar	gebäre	gebier	geboren
143	geben	gebe, gibst, gibt	gab	gäbe	gib	gegeben
144	gedeihen	gedeih/e, ∼ st, ∼ t	gedieh	gediehe	gedeih(e)	gediehen
145	gehen	geh/e, ∼ st, ∼ t	ging, ∼ est	ginge	geh(e)	gegangen
146	gelingen	es gelingt	es gelang	es gelänge	geling(e)	gelungen
147	gelten	gelte, giltst, gilt	galt, ∼ (e)st	gölte (gälte)	gilt	gegolten
148	genesen	genes/e, ∼ (es)t, ∼ t	genas, ∼ est	genäse	genese	genesen
149	genießen	genieß/e, ∼ (es)t, ∼ t	genoß, genossest	genösse	genieß(e)	genossen
150	geschehen	es geschieht	es geschah	es geschähe	—	geschehen
151	gewinnen	gewinn/e, ∼ st, ∼ t	gewann, ∼ (e)st	gewönne (gewänne)	gewinn(e)	gewonnen
152	gießen	gieß/e, ∼ (es)t, ∼ t	goß, gossest	gösse	gieß(e)	gegossen
153	gleichen	gleich/e, ∼ (e)st, ∼ t	glich, ∼ (e)st	gliche	gleich(e)	geglichen
154	gleißen	gleiß/e, ∼ (es)t, ∼ t	gleißte (gliß),	glisse	gleiß(e)	gegleißt
155	gleiten	gleit/e, ∼ est, ∼ et	glitt, ∼ (e)st	glitte	gleit(e)	geglitten
156	glimmen	glimme/e, ∼ st, ∼ t	glomm, (glimmte)	glömme	glimme	geglommen
157	graben	grabe, gräbst, gräbt	grub, ∼ (e)st	grübe	grab(e)	gegraben
158	greifen	greif/e, ∼ st, ∼ t	griff, ∼ (e)st	griffe	greif(e)	gegriffen
159	haben	habe, hast, hat	hatte	hätte	hab(e)	gehabt
160	halten	halte, hältst, hält	hielt, ∼ (e)st	hielte	halt(e)	gehalten
161	hängen	hänge, hängst, hängt	hing, ∼ (e)st	hinge	häng(e)	gehangen
162	hauen	hau/e, ∼ st, ∼ t	hieb (haute)	hiebe	hau(e)	gehauen
163	heben	heb/e, ∼ st, ∼ t	hob (hub), ∼ (e)st	höbe (hübe)	heb(e)	gehoben
164	heißen	heiße, ∼ (es)t, ∼ t	hieß, ∼ est	hieße	heiß(e)	geheißen
165	helfen	helfe, hilfst, hilft	half, ∼ (e)st	hülfe	hilf	geholfen

	Infiniv	Indicativ Präsens	Indicativ Präteritum	Konjunktiv Präteritum	Imperativ	Partizip des Perfekts
166	kennen	kenn/e, ∼ st, ∼ t	kannte	kennte	kenn(e)	gekannt
167	klimmen	klimm/e, ∼ st, ∼ t	klomm, ∼ (e)st	klömme	klimm(e)	geklommen
168	klingen	kling/e, ∼ st, ∼ t	klang, ∼ (e)st	klänge	kling(e)	geklungen
169	kneifen	kneif/e, ∼ st, ∼ t	kniff	kniffe	knief(e)	gekniffen
170	kommen	komm/e, ∼ st, ∼ t	kam	käme	komm(e)	gekommen
171	können	kann, ∼ st, ∼ , können	konnte	könnte	—	gekonnt
172	kriechen	kriech/e, ∼ st, ∼ t	kroch	kröche	kriech(e)	gekrochen
173	laden	lad/e, ∼ est (lädst), ∼ et (lädt)	lud (ladete), ∼ (e)st	lüde (ladete)	lad(e)	geladen
174	lassen	lasse, lässest (läßt), läßt	ließ, ∼ est	ließe	laß(lasse)	gelassen
175	laufen	laufe, läufst, läuft	lief, ∼ (e)st	liefe	lauf(e)	gelaufen
176	leiden	leid/e, ∼ est, ∼ et	litt, ∼ (e)st	litte	leid(e)	gelitten
177	leihen	leih/e, ∼ st, ∼ t	lieh, ∼ (e)st	liehe	leih(e)	geliehen
178	lesen	lese, lies(es)t, liest	las, ∼ est	läse	lies	gelesen
179	liegen	lieg/e, ∼ st, ∼ t	lag	läge	liege	gelegen
180	lügen	lüg/e, ∼ st, ∼ t	log, ∼ (e)st	löge	lüg(e)	gelogen
181	meiden	meid/e, ∼ est, ∼ et	mied, ∼ (e)st	miede	meid(e)	gemieden
182	melken	melk/e, ∼ st (milkst), ∼ t (milkt)	melkte (molk)	mölke	melk(e)	gemelkt gemolken
183	messen	messe, missest, (mißt), mißt	maß, ∼ est	mäße	miß	gemessen
184	mißlingen	es mißlingt	es mißlang	es mißlänge	—	mißlungen
185	mögen	mag, ∼ st, ∼ , mögen	mochte	möchte	—	gemocht
186	müssen	muß, ∼ t, ∼ , müßen, müßt (müsset), müssen	mußte	müßte	—	gemußt
187	nehmen	nehme, nimmst, nimmt	nahm, ∼ (e)st	nähme	nimm	genommen
188	nennen	nenn/e, ∼ st, ∼ t	nannte	nennte	nenn(e)	genannt
189	pfeifen	pfeif/e, ∼ st, ∼ t	pfiff, ∼ (e)st	pfiffe	pfeif(e)	gepfiffen
190	pflegen	pfleg/e, ∼ st, ∼ t	pflegte (pflog), ∼ st	pflegte (pflöge)	pfleg(e)	gepflogen
191	preisen	preis/e, ∼ (es)t, ∼ t	pries, ∼ est	priese	preis(e)	gepriesen
192	quellen	quelle, quillst (quellst), quillt (quellt)	quoll (quellte)	quölle	quill (quelle)	gequollen (gequellt)
193	raten	rate, rätst, rät	riet, ∼ (e)st	riete	rat(e)	geraten
194	reiben	reib/e, ∼ st, ∼ t	rieb, ∼ (e)st	riebe	reib(e)	gerieben
195	reißen	reiß/e, ∼ (es)t, ∼ et	riß, rissest	risse	reiß(e)	gerissen
196	reiten	reit/e, ∼ est, ∼ et	ritt, ∼ (e)st	ritte	reit(e)	geritten
197	rennen	renn/e, ∼ st, ∼ t	rannte	rennte	renn(e)	gerannt
198	riechen	riech/e, ∼ st, ∼ t	roch	röche	riech(e)	gerochen

	Infiniv	Indicativ Präsens	Indicativ Präteritum	Konjunktiv Präteritum	Imperativ	Partizip des Perfekts
199	ringen	ring/e, ∼ st, ∼ t	rang	ränge	ring(e)	gerungen
200	rinnen	rinn/e, ∼ st, ∼ t	rann, ∼ (e)st	ränne (rönne)	rinn(e)	geronnen
201	rufen	ruf/e, ∼ st, ∼ t	rief, ∼ (e)st	riefe	ruf(e)	gerufen
202	saufen	saufe, säufst, säuft	soff, ∼ (e)st	söffe	sauf(e)	gesoffen
203	saugen	saug/e, ∼ st, ∼ t	sog (saugte), ∼ (e)st	söge	saug(e)	gesogen (gesaugt)
204	schaffen	schaff/e, ∼ st, ∼ t	schuf, (schaffte), ∼ (e)st	schüfe	schaff(e)	geschaffen (geschafft)
205	schallen	schall/e, ∼ st, ∼ t	schallte (scholl)	schallete (schölle)	schall(e)	geschollen (geschallt)
206	scheiden	scheid/e, ∼ est, ∼ et	schied, ∼ (e)st	schiede	scheid(e)	geschieden
207	scheinen	schien/e, ∼ st, ∼ t	schien, ∼ (e)st	schiene	schein(e)	geschienen
208	schelten	schelt/e, ∼ schiltst, ∼ schilt	schalt, ∼ (e)st	schölte	schilt	gescholten
209	scheren	schere, schierst (scherst), schiert (schert)	schor (scherte)	schöre	schier, scher(e)	geschoren
210	schieben	schieb/e, ∼ st, ∼ t	schob, ∼ (e)st	schöbe	schieb(e)	geschoben
211	schießen	schieß/e, ∼ (es)t, ∼ t	schoß, schossest	schösse	schieß(e)	geschossen
212	schinden	schind/e, ∼ est, ∼ et	schund, ∼ (e)st	schünde	schind(e)	geschunden
213	schlafen	schlafe, schläfst, schläft	schlief, ∼ (e)st	schliefe	schlaf(e)	geschlafen
214	schlagen	schlage, schlägst, schlägt	schlug, ∼ (e)st	schlüge	schlag(e)	geschlagen
215	schleichen	schleich/e, ∼ st, ∼ t	schlich, ∼ (e)st	schliche	schleich(e)	geschlichen
216	schleifen	schleif/e, ∼ st, ∼ t	schliff, ∼ (e)st	schliffe	schleif(e)	geschliffen
217	schleißen	schleiß/e, ∼ (es)t, ∼ t	schliß(schleißte), schlissest	schlisse	schleiß(e)	geschlissen
218	schließen	schließe, ∼ (es)t, ∼ t	schloß, schlossest	schlösse	schließ(e)	geschlossen
219	schlingen	schling/e, ∼ st, ∼ t	schlang, ∼ (e)st	schlänge	schling(e)	geschlungen
220	schmeißen	schmeiß/e, ∼ (es)t, ∼ t	schmiß, schmissest	schmisse	schmeiß(e)	geschmissen
221	schmelzen	schmelze, schmilz(es)t, schmilzt	schmolz (schmelzte) ∼ est	schmölze	schmilz	geschmolzen (geschmelzt)
222	schnauben	schnaub/e, ∼ st, ∼ t	schnaubte (schnob)	schnaubte (schnöbe)	schnaub(e)	geschnaubt (geschnoben)
223	schneiden	schneid/e, ∼ est, ∼ et	schnitt, ∼ (e)st	schnitte	schneid(e)	geschnitten
225	schrecken	schrecke, schrickst, (schreckst), schrickt (schreckt)	schrak, ∼ (e)st (schreckte)	schräke (schreckte)	schrick (schrecke)	erschrocken
226	schreiben	schreib/e, ∼ st, ∼ t	schrieb, ∼ (e)st	schriebe	schreib(e)	geschrieben
227	schreien	schrei/e, ∼ st, ∼ t	schrie	schriee	schrei(e)	geschrie(e)en

	Infiniv	Indicativ Präsens	Indicativ Präteritum	Konjunktiv Präteritum	Imperativ	Partizip des Perfekts
228	schreiten	schreit/e, ∼ est, ∼ et	schritt, ∼ (e)st	schritte	schreit(e)	geschritten
229	schweigen	schweig/e, ∼ st, ∼ t	schwieg, ∼ (e)st	schwiege	schweig(e)	geschwiegen
230	schwellen	schwelle, schwillst, (schwellst) schwillt (schwellt)	schwoll, ∼ (e)st (schwellte)	schwölle (schwellte)	schwill (schwelle)	geschwollen (geschwellt)
231	schwimmen	schwimm/e, ∼ st, ∼ t	schwamm, ∼ (e)st	schwömme (schwämme)	schwimm(e)	geschwommen
232	schwinden	schwind/e, ∼ est, ∼ et	schwand, ∼ (e)st	schwände	schwind(e)	geschwunden
233	schwingen	schwing/e, ∼ st, ∼ t	schwang, ∼ (e)st	schwänge	schwing(e)	geschwungen
234	schwören	schwör/e, ∼ st, ∼ t	schwur, (schwor), ∼ (e)st	schwüre	schwöre	geschworen
235	sehen	sehe, siehst, sieht	sah, ∼ st	sähe	sieh(e)	gesehen
236	sein	bin, bist, ist, sind, seid, sind	war, ∼ st	wäre	sei, seid	gewesen
237	senden	send/e, ∼ est, ∼ et	sandte (sendete), ∼ st	sendete	send(e)	gesandt (gesendet)
238	sieden	sied/e, ∼ est, ∼ et	sott (siedete), ∼ (e)st	sötte,(siedete)	sied(e)	gesotten (gesiedet)
239	singen	sing/e, ∼ st, ∼ t	sang, ∼ (e)st	sänge	sing(e)	gesungen
240	sinken	sink/e, ∼ (e)st, ∼ t	sank, ∼ (e)st	sänke	sink(e)	gesunken
241	sinnen	sinn/e, ∼ st, ∼ t	sann, ∼ (e)st	sänne (sönne)	sinn(e)	gesonnen
242	sitzen	sitz/e, ∼ (e)st, ∼ t	saß, ∼ est	säße	sitze	gesessen
243	sollen	soll, ∼ st	sollte	sollte	—	gesollt
244	speien	spei/e, ∼ st, ∼ t	spie	spiee	spei(e)	gespie(e)n
245	spinnen	spinn/e, ∼ st, ∼ t	spann, ∼ (e)st	spönne (spänne)	spinn(e)	gesponnen
246	sprechen	spreche, sprichst, spricht	sprach, ∼ (e)st	spräche	sprich	gesprochen
247	sprießen	sprieß/e, ∼ (es)t, ∼ t	sproß, sprossest	sprösse	sprieß(e)	gesprossen
248	springen	spring/e, ∼ st, ∼ t	sprang, ∼ (e)st	spränge	spring(e)	gesprungen
249	stechen	steche, stichst, sticht	stach, ∼ (e)st	stäche	stich	gestochen
250	stecken	steck/e, ∼ st, ∼ t	stak (steckte)	stäke (steckte)	steck(e)	gesteckt
251	stehen	steh/e, ∼ st, ∼ t	stand, ∼ (e)st	stände (stünde)	steh(e)	gestanden
252	stehlen	stehle, stiehlst, stiehlt	stahl	stöhle (stähle)	stiehl	gestohlen
253	steigen	steig/e, ∼ st, ∼ t	stieg, ∼ (e)st	stiege	steig(e)	gestiegen
254	sterben	sterbe, stirbst, stirbt	starb	stürbe	stirb	gestorben
255	stieben	stieb/e, ∼ st, ∼ t	stob, ∼ (e)st	stöbe	stieb(e)	gestoben
256	stinken	stink/e, ∼ st, ∼ t	stank, ∼ (e)st	stänke	stink(e)	gestunken

	Infiniv	Indicativ Präsens	Indicativ Präteritum	Konjunktiv Präteritum	Imperativ	Partizip des Perfekts
257	stoßen	stoße, stöß(es)t, stößt	stieß, ∼ est	stieße	stoß(e)	gestoßen
258	streichen	streich/e, ∼ st, ∼ t	strich, ∼ (e)st	striche	streich(e)	gestrichen
259	streiten	steit/e, ∼ est, ∼ et	stritt, ∼ (e)st	stritte	streit(e)	gestritten
260	tragen	trage, trägst, trägt	trug	trüge	trag(e)	getragen
261	treffen	treffe, triffst, trifft	traf, ∼ (e)st	träfe	triff	getroffen
262	treiben	treib/e, ∼ st, ∼ t	trieb	triebe	treib(e)	getrieben
263	treten	trete, trittst, tritt	trat, ∼ (e)st	träte	tritt	getreten
264	triefen	trief/e, ∼ st, ∼ t	troff (triefte), ∼ (e)st	tröffe (triefte)	trief(e)	getroffen (getrieft)
265	trinken	trink/e, ∼ st, ∼ t	trank, ∼ (e)st	tränke	trink(e)	getrunken
266	trügen	trüg/e, ∼ st, ∼ t	trog, ∼ (e)st	tröge	trüg(e)	getrogen
267	tun	tue, tust, tut, tun	tat, ∼ (e)st	täte	tu(e)	getan
268	verderben	verderbe, verdirbst, verdirbt	verdarb	verdürbe	verdirb	verdorben verderbt
269	verdrießen	verdrieß/e, ∼ (es)t, ∼ t	verdroß, verdrossest	verdrösse	verdrieß(e)	verdrossen
270	vergessen	vergesse, vergissest (vergißt), vergißt	vergaß, ∼ est	vergäße	vergiß	vergessen
271	verlieren	verlier/e, ∼ st, ∼ t	verlor	verlöre	verlier(e)	verloren
272	wachsen	wachse, wächs(es)t, wächst	wuchs, ∼ est	wüchse	wachs(e)	gewachsen
273	wägen	wäg/e, ∼ st, ∼ t	wog (wägte)	wöge (wägte)	wäg(e)	gewogen (gewägt)
274	waschen	wasche, wäsch(e)st, wäscht	wusch, ∼ (e)st	wüsche	wasch(e)	gewaschen
275	weben	web/e, ∼ st, ∼ t	webte(wob, wobest)	webte(wöbe)	web(e)	gewebt (gewoben)
276	weichen	weich/e, ∼ st, ∼ t	wich, ∼ est	wiche	weich(e)	gewichen
277	weisen	weis/e, ∼ (es)t, ∼ t	wies, ∼ est	wiese	weis(e)	gewiesen
278	wenden	wend/e, ∼ est, ∼ et	wandte (wendete)	wendete	wende	gewandt (gewendet)
279	werben	werbe, wirbst, wirbt	warb	würbe	wirb	geworben
280	werden	werde, wirst, wird	wurde (ward)	würde	werd(e)	geworden
281	werfen	werfe, wirfst, wirft	warf, ∼ (e)st	würfe	wirf	geworfen
282	wiegen	wieg/e, ∼ st, ∼ t	wog	wöge	wieg(e)	gewogen
283	winden	wind/e, ∼ est, ∼ et	wand, ∼ (e)st	wände	wind(e)	gewunden
284	wissen	weiß, ∼ t, ∼ ;wissen wißt, wissen	wußte	wüßte	wisse	gewußt
285	wollen	will, ∼ st, ∼ , wollen	wollte	wollte	wolle	gewollt
286	zeihen	zeih/e, ∼ st, ∼ t	zieh, ∼ (e)st	ziehe	zeih(e)	geziehen

Table of conjugations of German verbs

	Infiniv	Indicativ Präsens	Indicativ Präteritum	Konjunktiv Präteritum	Imperativ	Partizip des Perfekts
287	ziehen	zieh/e, ∼ st, ∼ t	zog, ∼ (e)st	zöge	zieh(e)	gezogen
288	zwingen	zwing/e, ∼ st, ∼ t	zwang, ∼ (e)st	zwänge	zwing(e)	gezwungen
289	scheißen	scheiß/e, ∼ (es)t, ∼ t	schiß, ∼ ssest	schisse	scheiße	geschissen
290	spleißen	spleiß/e, ∼ (es)t, ∼ t	spliß, ∼ ssest	splisse	spleiße	gesplissen
291	wringen	wring/e, ∼ st, ∼ t	wrang	wränge	wring(e)	gewrungen
292	küren	kür/e, ∼ (e)st, ∼ t	kor	köre	kür(e)	gekoren
293	salzen	salz/e, ∼ t, ∼ t	salzt/e, ∼ est, ∼ e	salzte	salz(e)	gesalzt
294	mahlen	mahl/e, ∼ st, ∼ t	mahlt/e, ∼ est	mahlte	mahl(e)	gemahlen
295	spalten	spalt/e, ∼ est, ∼ et	spaltet/e, ∼ est, ∼ e	spaltete	spalt(e)	gespalten
296	verlöschen	verlösche, verlisch(e)st, verlischt	verlosch, ∼ est	verlösche	verlisch	verloschen
297	verbleichen	verbleich/e, ∼ st, ∼ t	verbleichte (verblich)	verbleichte (verbliche)	verbleich(e)	verbleicht (verblichen)

Table of conjugations of German verbs

Code	Case	Maskuline	Feminine	Neuter
S0	Pluralia Tantum			
S1	Nom.	der Wald	—	das Brot
	Gen.	des Wald(e)s	—	des Brot(e)s
	Dat.	dem Wald(e)	—	dem Brot(e)
	Acc.	den Wald	—	das Brot
S2	Nom.	der Bär	—	—
	Gen.	des Bär(e)n	—	—
	Dat.	dem Bär(e)n	—	—
	Acc.	den Bär(e)n	—	—
S3	Nom.	—	die Bar	—
	Gen.	—	der Bar	—
	Dat.	—	der Bar	—
	Acc.	—	die Bar	—
S4	Nom.	der Bus	—	das Zeugnis
	Gen.	des Busses	—	des Zeugnisses
	Dat.	dem Bus	—	dem Zeugnis
	Acc.	den Bus	—	das Zeugnis
S5	Nom.	der Buchstabe	—	—
	Gen.	des Buchstabens	—	—
	Dat.	dem Buchstaben	—	—
	Acc.	den Buchstaben	—	—
S6	Nom.	—	—	das Herz
	Gen.	—	—	des Herzens
	Dat.	—	—	dem Herzen
	Acc.	—	—	das Herz

Table Of Flections of German Nouns

Code	Case	Pluralforms
P0	Singularia Tantum	
P1	Nom.	die Stoffe
	Gen.	der Stoffe
	Dat.	den Stoffen
	Acc.	die Stoffe
P1U	Nom.	die Bäume
	Gen.	der Bäume
	Dat.	den Bäumen
	Acc.	die Bäume
P2	Nom.	die Esel
	Gen.	der Esel
	Dat.	den Eseln
	Acc.	die Esel
P2U	Nom.	die Äpfel
	Gen.	der Äpfel
	Dat.	den Äpfeln
	Acc.	die Äpfel
P3	Nom.	die Bauern
	Gen.	der Bauern
	Dat.	den Bauern
	Acc.	die Bauern
P4	Nom.	die Felder
	Gen.	der Felder
	Dat.	den Feldern
	Acc.	die Felder
P4U	Nom.	die Dächer
	Gen.	der Dächer
	Dat.	den Dächern
	Acc.	die Dächer
P5	Nom.	die Autos
	Gen.	der Autos
	Dat.	den Autos
	Acc.	die Autos

Table Of Flections of German Nouns

Code	Case	Pluralforms
P6	Nom.	die Reifen
	Gen.	der Reifen
	Dat.	den Reifen
	Acc.	die Reifen
P6	Nom.	die Öfen
	Gen.	der Öfen
	Dat.	den Öfen
	Acc.	die Öfen
P7	Nom.	die Freundinnen
	Gen.	der Freundinnen
	Dat.	den Freundinnen
	Acc.	die Freundinnen
P8	Nom.	die Geheimnisse
	Gen.	der Geheimnisse
	Dat.	den Geheimnissen
	Acc.	die Geheimnisse
P9	Nom.	die Maxima
	Gen.	der Maxima
	Dat.	den Maxima
	Acc.	die Maxima
P10	Nom.	die Gymnasien
	Gen.	der Gymnasien
	Dat.	den Gymnasien
	Acc.	die Gymnasien
P11	Other words	

Table Of Flections of German Nouns

Code	Example
0G0000000	Er ist der Lehrer.
EG0000000	Es wird Sommer.
0L0000000	Ich bleibe hier.
0T0000000	Du darfst morgen bleiben.
0M0000000	Der Schrank ist aus Eichenholz.
0C0000000	Er bleibt wegen des Festivals.
0U0000000	Die Summe bleibt zur Verfügung.
0000N0000	Das Buch gehört mir.
00000N000	Wir gedenken des 40. Jahrestags der Verkündung des Grundgesetzes.
0Z0000000	Er scheint abzureisen.
000000000	Der Mann weint.
E00000000	Es schneit.
00n000000	Er gewinnt (die Wette).
0000n0000	Das gelingt (mir).
00000n000	Er starb (eines qualvollen Todes).
000000p00	Er antwortete (auf die Frage).
00000000A	Er kommt mit dem Zug\morgen\hier.
00000000L	Er kommt hier.
00000000T	Er kommt morgen.
00000000M	Der Bau des Schiffes ist schon weit gediehen.
00000000C	Der arme Mann raste vor Schmerzen.
00000000U	Er fühlte nach dem Schalter im Dunkeln.
00000000S	Wir haben das Feuer mit Holz gefeuert.
00000000O	Die Firma handelt mit den Chinesen.
00000000R	Dieses Gerät gilt als das Beste auf diesem Gebiet.
00I000000	Jeder konnte dabeisein.
00Z000000	Was hat das zu bedeuten.
00N000000	Er bekommt kein Geschenk.
E0N000000	Auf dieser Strecke fährt es sich gut.
00N0n0000	Ich zünde die Kerze (mit dem Feuerzeug) an.
00N00n000	Man hat ihn (des Mordes) beschuldigt.
00N000p00	Der Mann versuchte mich (zu diesem Glauben) zu bekehren.
00Ni00000	Ich hörte ihn die ganze Nacht (schnarchen).
00Nz00000	Die hübsche junge Dame forderte ihn auf (teilzunehmen).
00N00000A	Ich kann mich (an diesem Ort) nicht gut zurechtfinden.
00N00000L	Der Patient wurde (aus dem Krankenhaus) entlassen.
00N00000T	Der Laden ist bis fünf Uhr geöffnet.
00N00000M	Ich glaube nicht, daß er mich hoch eingschätzt hat.
00N00000C	Er überschlug sich fast vor Diensteifer.

Code	Example
00N00000U	Bei der Military hat schon mancher Reiter ein Pferd zu Tode geritten.
00N00000S	Mit diesen Daten kann ich nichts anfangen.
00N00000O	Ich habe mich viele Jahre mit ihm geschrieben.
00N00000R	Dadurch hat man ihn als einen Versager eingeschätzt.
00NN00000	Das habe ich mich schon oft gefragt.
00N0N0000	Er hat ihm diese Geschichte eingeflüstert.
00N0N000L	Vor Verzweiflung hat er sich eine Kugel durch den Kopf gejagd.
00N0N000M	Er hat sich in Nijmegen beim Wandern die Füße wund gelaufen.
00N0N000C	Ich verspreche mir viel von dieser Behandlung.
00N0N000U	Wenn du hier arbeiten willst solltest du dir dies zu eigen machen.
00N00N000	Jetzt ist er aller Sorgen enthoben.
00N000P00	Man konnte erwarten, daß sie sich gegen ihn aufbäumen würde.
00N000P0M	Ich glaube, daß er sich positiv zu diesem Vorschlag stellt.
00NI00000	Er lehrt ihn schreiben.
00NZ00000	Sie lehrte ihn Gedichte zu schreiben.
0000N0000	Bleibe mit den Fingern von Sachen, die dir nicht gehören.
E000N0000	Es geht mir schon viel besser.
00n0N0000	Damals opferte man den Göttern noch eine Ziege oder eine Kuh.
0000N0000	Er hat der Versammlung beigewohnt.
0000N0p00	Wir möchten ihm (zum Geburtstag) gratulieren.
0000N000L	Er half dem Behinderten in den Wagen.
0000N000M	Wenn du so etwas Dummes getan hast, geschieht dir so ein Schicksal recht.
0000N000S	Zuhause werden wir ihm mit Blumen und Geschenken aufwarten.
0000N0P00	Sein Hobby geht ihm über alles andere.
00Z0N0000	Beliebt es ihm heute noch Besuch zu empfangen?
00000N000	Ich kann deiner Hilfe nicht entraten.
E0000N000	Diese Lösung ist so logisch, daß es keiner Erklärung braucht.
00000N00O	Weil ich mir nicht sicher war, pflegte ich Rats mit ihm.
0000NN000	Weil ich mir nicht sicher war, erholte ich mir Rates bei ihm.
000000P00	Ich kann nicht für ihre Sicherheit einstehen.
E00000P00	Mit uns ist es auf dieser Reise gutgegangen.
00n000P00	Ich möchte (dich) auf diese Gefahr hinweisen.
0000n0P00	Er wollte (mir) nicht zu diesem Kauf raten.
000000P0L	Bei dem Fall haute er mit den Kopf auf die Straße.
000000P0M	Er trug eine Krawatte die gut zu dem Anzug aussah.
000000PP0	Sie ist mit dem Antrag an ihn herangetreten.
???????????	

Table Of Verbal Complementation Codes

INDEX

Special characters

A

B

C

D

E

F

G

H

I

L

M

N

O

P

R

S

T

U

V

W

X

Z