J. Linguistics **35** (1999), 1–42. Printed in the United Kingdom

German noun inflection[1]

LYNNE CAHILL AND GERALD GAZDAR

University of Sussex

(Received 13 October 1997; revised 20 April 1998)

This is the second of a series of three papers that, taken together, will give an essentially complete account of inflection in standard German. In this paper we present that part of the account that covers nouns, one that captures all the regularities, subregularities and irregularities that are involved, but with a focus on the subregularities. Inflected forms are defined in terms of their syllable structure, as proposed in Cahill (1990a, b, 1993). The analysis is formulated as a DATR theory – a set of lexical axioms – from which all the relevant facts follow as theorems. DATR is a widely used formal lexical knowledge representation language developed for use in computational linguistics.

1. Introduction

This paper is the successor to Cahill & Gazdar (1997), a paper that presents an analysis of adjective, determiner and pronoun inflection in German. Our general approach to inflectional morphology falls within the tradition that treats paradigms (inflectional classes, declensions, conjugations, etc.) as analytically central[2] rather than epiphenomenal or of secondary status.[3] The central notion is the lexeme, not the word or the morpheme. Words exist, but only as REALIZATIONS of (morphosyntactic specifications of) lexemes – hence Stump's use of the term REALIZATIONAL to characterize this tradition.

[1] The present paper is a direct descendant of a talk ('The lexical representation of German morphology') given at the Autumn Meeting of the Linguistics Association of Great Britain at UMIST in Manchester on 22nd September 1989 (the fragment as presented there is included in Roger Evans & Gerald Gazdar (eds.) (1990). *The* DATR *papers*. University of Sussex: Cognitive Science Research Paper CSRP 139. 85–93.). We thank members of the audience for their comments on that occasion. And we thank those attending the June 1996 ESRC *Frontiers of Research in Morphology* Seminar at Surrey, especially Grev Corbett and Andy Spencer, for their comments on an oral presentation of a much more recent version of the analysis. We are grateful to Jim Kilbury for his detailed comments on the May 1995 edition of the DATR fragment. And we are grateful to our two *JL* referees, to Jim Kilbury and to Dieter Wunderlich for their comments on the original manuscript of this paper. This research is supported by a grant *Multilingual lexical knowledge representation*, (R000235724) to the authors from the ESRC (UK).

[2] As in the work of Matthews (1972), van Marle (1985), Zwicky (1985, 1990), Carstairs (1987) and Stump (e.g. 1992; 1993a, b, c; 1995).

[3] Thus, for example, inflectional class is a secondary notion for Wurzel (1990: 204): for him it is the citation form that determines the inflectional class, not the converse.

Morphemes also exist, but only as second class citizens. The appearance of a morpheme is just one among several ways that morphosyntactic information gets expressed in the realization of a lexeme as a word (see Wurzel 1990: 208–209). And we share Zwicky's view that 'all realization rules are treated as expressing DEFAULTS, which are automatically overridden by more specific rules (and these in turn by still more specific rules, and so on)' (1985: 372).

As regards current work, our approach is closely related to Corbett & Fraser's Network Morphology[4] and the way in which we represent nominal inflection in German is similar to that proposed for Russian nominal inflection in Corbett & Fraser (1993). It is also closely related to the most recent version of Stump's Paradigm Function Morphology. The present paper is entirely about inflectional morphology but, in our approach, unlike those of Stump and Corbett et al., abstract inflectional rules are typically stated in terms of phonological units, most commonly the syllable and the segment (as in Cahill 1990a, b; 1993). Gibbon and his collaborators in the ILEX (Integrated Lexicon with EXceptions) project at Bielefeld[5] have pioneered the use of default inheritance hierarchies for the representation of lexical phonology and morphophonology. Our work is thus also indebted to theirs.

Much of the recent published work on German noun inflection[6] has focussed on the phonological and morphophonological issues that arise in an analysis, rather than on the inflectional system per se. There is a good deal of discussion of the nature of umlaut and this will be considered in relation to our own approach in section 4.2 below. Another issue which has attracted attention in the literature is the role of schwa in the inflection of German. Although we provide formal accounts of final consonant devoicing, schwa and umlaut, they are not the focus of the present enterprise. We have no choice but to provide analyses since we are engaged in giving a fully explicit, fully axiomatised theory of German noun inflection. But their status is essentially modular – given an alternative, but descriptively equivalent, theory of the phonology of umlaut, say, it ought to be possible to use it to replace the one we give without any significant consequences for the rest of our analysis.

[4] See Brown et al. (1996), Brown & Hippisley (1994) and Fraser & Corbett (1995, 1997) for work in this framework.

[5] See Bleiching (1992, 1994), Bleiching et al. (1996), Gibbon (1990, 1992), Gibbon & Bleiching (1991), Reinhard (1990) and Reinhard & Gibbon (1991) for examples of this work.

[6] The literature on German noun inflection is large and includes such milestones as Wurzel (1970), Lieber (1981) and Wurzel (1984). A systematic review that did justice to the relevant literature of the last thirty years would be at least as long as the present paper. For the most part, we restrict our attention below to work on German noun inflection in the inheritance/realizational tradition published in the decade that began with Carstairs (1987).

The work on German nouns which perhaps most closely resembles that described here is that by Bleiching (1992, 1994). Bleiching provides accounts of German noun and verb inflection in a framework which is broadly similar to ours, although she concentrates on stress in compound nouns and does not provide a full account of the non-affixal alternations. Also very closely related is the analysis of German noun inflection given by Reinhard (1990), Reinhard & Gibbon (1991) and Gibbon (1992) which concentrates on umlaut. In common with this Bielefeld work, we make pervasive use of syllable addresses to specify phonological structure, and we allow such addresses to be extended with morphosyntactic attributes (as in Gibbon's use of `<orth peak vowel plural>` (1992: 48)).

2. THE DATR LANGUAGE

We implement and present our theory of German inflection in the lexical knowledge representation language DATR (see Evans & Gazdar 1996; Keller 1995, 1996). DATR is a rather spartan non-monotonic language for defining inheritance networks with path-value equations. The development of DATR was guided by a number of concerns which we summarise here. The objective was to design a language which (i) has an explicit theory of inference, (ii) has an explicit declarative semantics, (iii) can be readily and efficiently implemented,[7] (iv) has the necessary expressive power to encode the lexical information presupposed by work in the unification grammar tradition, and (v) can express all the evident generalizations and subgeneralizations about such entries. In keeping with its intendedly minimalist character, it lacks many of the constructs embodied either in general purpose AI knowledge representation languages or in contemporary grammar formalisms. The language is nonetheless sufficiently expressive to represent concisely the structure of lexical information at a variety of domains of language description.

[7] There have been more than a dozen different implementations of the DATR language. They include Evans's (Brighton) implementation, which is written in Prolog and runs on most Unix platforms; Gibbons's (Bielefeld) DDATR Scheme, NODE Sicstus Prolog, ZDATR C, and `awk` implementations; Kilbury's (Düsseldorf) QDATR Arity, Quintus and Sicstus Prolog implementations; and Illouz's (Paris) implementation of CDATR (in C). All of these are freely available on request, as is an extensive archive of over two hundred example fragments some of which illustrate formal techniques and others of which are applications of DATR to the lexical phonology, morphology, syntax or semantics of a wide variety of different languages (including nontrivial fragments of aspects of the lexicons of Arabic, Czech, Dakota, English, French, German, Gikuyu, Italian, Japanese, Latin, Polish, Portuguese, Russian and Spanish, and smaller indicative fragments for Baoule, Dan, Dutch, Hua, Nyanja, Serbo-Croat, Swahili, Tem and Welsh Romany. The URL `http://www.cogs.sussex.ac.uk/lab/nlp/datr/datr.html` and anonymous FTP to `ftp.cogs.sussex.ac.uk` and directory `/pub/nlp/DATR` provide access to various DATR implementations, the example archive, and some relevant papers and documentation.

It should be stressed that DATR itself is no more than a very general LANGUAGE for lexical description and therefore does not commit or restrict the linguist using it to any particular linguistic framework, theory or formalism, nor is it restricted in the class of natural languages that it can be used to describe. Clearly, it is well suited to lexical frameworks that embrace or are consistent with inheritance and non-monotonicity through networks of nodes, but these are not requirements. DATR can be (and has been) used to implement differing theoretical approaches (including ILEX, HPSG (Head-driven Phrase Structure Grammar), Word Grammar, LTAG (Lexicalized Tree-adjoining Grammar), Finite State Morphology, Network Morphology, Paradigm Function Morphology), and is perhaps best thought of as a programming language which can be used to implement and test linguistic theories. Indeed, it would not be entirely misleading to think of DATR as a kind of assembly language for constructing (or reconstructing) higher level theories of lexical representation. Unlike most other formal languages proposed for lexical knowledge representation, DATR is also not restricted in the domains of linguistic description to which it can sensibly be applied. It is designed to be equally applicable at phonological, orthographic, morphological, syntactic and semantic domains of description. But it is not intended to replace existing approaches to those domains. DATR cannot be (sensibly) used without a prior decision as to the theoretical frameworks in which the description is to be conducted; there is thus no 'default' framework for describing, say, morphological facts in DATR.

In DATR, information is organised as a network of NODES, where a node is essentially just a collection of related information. In the context of lexical description, a node might correspond to a phoneme, a syllable, a morpheme, a word, a lexeme, etc., or a class of such items. For example, we might have a node describing an abstract `Word` in German, a node for the class of German nouns, a node for the subclass of German nouns that mark plurals with *-s*, a node for the particular noun lexeme `Klub` ('club') and still more for the individual words that are instances of this lexeme *Klub*, *Klub-s*. Each node has associated with it a set of equations that define partial functions from PATHS to VALUES where paths and values are both sequences of ATOMS (which are primitive objects). Atoms in paths are sometimes referred to as ATTRIBUTES. The syntax and terminology of DATR, like its name and its minimalist philosophy, owes more than a little to that of the unification grammar language PATR (Shieber 1986).

3. PHONOLOGY

Our interest in phonology in the present paper is restricted to those aspects of phonological structure that are relevant to the description of German inflection. That includes syllable structure as characterized below but does not include any structure above the level of the syllable, such as metrical

structure. We include primary lexical stress, which is indicated by the presence or absence of a marker on the rhyme of a syllable. Unlike Wiese (1996a: 273–274), we assume that monosyllabic roots have stress. Wiese argues that, since stress is only relevant when contrasting the syllables of polysyllabic roots, then it makes no difference whether one chooses to say that monosyllabic roots are stressed or not. He chose to say that the stress value is undefined. We choose to say that they are stressed as this simplifies the definition of focus (see section 3.2 below). But, since the stress mark is non-contrastive in monosyllables, the effects are the same.

We also restrict ourselves to a segmental representation of the phonology. Our German phonological segment inventory is taken from CELEX (Baayen et al. 1995) and uses the SAM-PA (Speech Assessment Methods Phonetic Alphabet) machine-readable phonetic alphabet (Wells 1987). As one of us has shown in earlier work (Cahill 1993), the step from representing structures with segments to representing the same structures with full feature sets at each point in the tree is relatively simple. We have not taken that step here because it would not add anything to most of the present analysis but it would make our DATR code much harder to read. A featural level is helpful in defining certain phonological alternations in German such as final consonant devoicing which we discuss in more detail in section 3.3 below. The same applies to the morphophonological alternations vowel lengthening and umlaut, which are discussed in sections 4.1 and 4.2 below.

We assume throughout that a fully inflected form is simply a string of phonological segments. For our present purposes, there is no need for the implicit tree structure of phonological objects to be made manifest in the output. It is, however, a simple task to modify the rules we give so as to make the tree structure explicit in the way inflected forms are encoded.

3.1 *Syllable structure*

As in Cahill (1990b) and Bleiching (1992), we define syllabic structures by means of simple context-free phrase structure rules:

```
syllable    →onset rhyme
rhyme       →stress peak coda
coda        →body tail
disyllable  →syllable syllable
trisyllable→syllable syllable syllable
```

A syllable consists of an onset and a rhyme; a rhyme consists of a stress value, a peak and a coda; and a coda consists of a body and a tail. A disyllable consists of two syllables, and a trisyllable of three. We can use DATR to express these context-free phrase structure rules as follows:[8]

[8] Bleiching (1992) does not address the issue of polysyllabic roots explicitly. She provides only for monosyllables and disyllables with a final schwa syllable. She does not class this latter as a separate syllable at all, but rather as a 'coda extension'.

```
Syllable:
    <phn $yll form> == ''<phn $yll onset>'' ''<phn $yll rhyme>''
    <phn $yll rhyme> == Stress ''<phn $yll peak>'' ''<phn $yll
                                                        coda>''
    <phn $yll coda> == ''<phn $yll body>'' ''<phn $yll tail>''
    <phn root> == <phn syl1>
    < > == Null.

Disyllable:
    < > == Syllable
    <phn root> == <phn syl2> <phn syl1>.

Trisyllable
    < > == Syllable
    <phn root> == <phn syl1> <phn syl2> <phn syl1>.
```

Strictly speaking, the definition of `Syllable` given above is a rule schema, rather than a rule, since it makes crucial use of a variable `$yll` that ranges over attributes (`syl1, syl2, ...`) that denote syllable positions. Note also that the maximally unspecified path (`< >`) at the `Syllable` node is defined by reference to `Null` which always returns the empty sequence as its value. An `<onset>`, `<peak>` or `<coda>` which is left undefined at lower levels of the hierarchy will, as a consequence, end up as null.

The definitions of di- and trisyllables numbers the syllables from the right. This is a language-specific aspect of our analysis and reflects the fact that German morphology primarily involves suffixation. Reference to final syllables is thus more frequent than reference to the initial syllables and it is technically convenient to have a constant identifier (`syl1` here) for final syllables.

Given this set of axioms for syllabic structure, we can now use them to help define particular concrete (poly)syllables. Here, for example, is a possible definition for the monosyllabic *-es* suffix, realised phonologically as /@s/.

```
Suffix_es:
    < > == Syllable
    <phn syl1 peak> == @
    <phn syl1 coda> == s.
```

Likewise, a disyllabic word root such as *Tutor* can be specified in terms of the individual components of its two syllables:[9]

```
Tutor
    < > = Noun_L
    <phn root form> == Disyllable
    <phn syl2 onset> == t
    <phn syl2 peak-> == u:
    <phn syl1 onset> == t
    <phn syl1 peak-> == O
    <phn syl1 tail-> == r.
```

[9] Default information for a lexeme node like this comes from the declensional class node, in this case, `Noun_L`.

https://doi.org/10.1017/S0022226798007294 Published online by Cambridge University Press

From these node definitions, taken together with the axioms for syllable structure given above, we can now infer that:

```
Suffix_es:
    <phn root form> = @ s.
Tutor:
    <phn root form> = t u: t O r.
```

In the case of `Tutor`, the phonological root form emerges as the result of concatenating the `<phn syl2 form>` value and the `<phn syl1 form>` value as determined by the equations given at the `Syllable` and `Disyllable` nodes.

Crucially for our approach to inflection, the realisation of any component of the phonological structure can be determined by morphosyntactic features. The idiosyncratic noun *Medikus*, for example, which has the plural form *Medici*,[10] could be represented thus:

```
Medikus:
    < > = = Disyllable
    <phn syl1 onset> = = m
    <phn syl2 peak> = = e:
    <phn syl1 onset> = = d
    <phn syl1 peak> = = i:
    <phn syl1 coda> = = k
    <phn syl1 coda plur> = = ts.
```

This immediately yields the distinction between singular and plural roots that is required for this word:

```
Medikus:
    <phn root form sing> = m e: d i: k
    <phn root form plur> = m e: d i: ts.
```

3.2 *Stress and focus*

In our analysis, a root is a sequence of syllables. Exactly one of these syllables is the FOCUSSED syllable. The focussed syllable of words whose inflection involves a stem alternation, such as umlaut, will normally be the syllable in which the alternation occurs. Whether or not a stem alternation occurs, the focussed syllable is, by default, the stressed syllable.

Our treatment of stress is thus very easy to state. The rule below can be paraphrased as 'if a syllable is identical to the focussed syllable, then the rhyme of that syllable will include the stress mark segment ('^'), if it isn't it won't'.

```
Stress:
    <phn $yll rhyme> = = IF: <EQ: <$yll ''<phn root focus>''>
                         THEN  ^
                         ELSE Null>.
```

[10] CELEX has the plural form *Medizi*.

3.3 *Final consonant devoicing*

Final consonant devoicing applies to the final tail of roots which appear either uninflected or with inflectional suffixes which do not begin with a vowel. Wiese (1996a, 201–203) has suggested that this amounts to devoicing of all syllable-final consonants, since the addition of a vowel-initial suffix results in the tail of the root becoming an onset of the following syllable (as required by the maximal onset principle). In the case of root-internal syllable-final consonants, these are all unvoiced, but they are invariant, since there cannot be any situation in which they become syllable initial. Therefore the only actual alternation between voiced and voiceless consonants appears root-finally. Like Kloeke (1982: 30–32), we take the view that the lexical representation of root-internal syllable-final consonants always specifies the voiceless segment, since this is the only one which can ever appear.[11] The lexical representation of roots where there is an alternation specifies the voiced counterpart, thereby distinguishing such roots from those which have invariant voiceless consonants root-finally (e.g., to distinguish between *Rat* and *Rad*). The devoiced variants are then determined by checking the suffix of the form in question. If there is a suffix which begins with a vowel, then no devoicing takes place. If there is no suffix, or the suffix begins with a consonant then devoicing does take place. This is defined by the following equation (which forms part of our definition of the `Word` node):

```
Word:
  <phn syl1 tail> == IF:   <VOWEL:<''<mor suffix>''>
                     THEN           ''<phn syl1 tail->''
                     ELSE Devoice:<''<phn syl1 tail->''>>.
```

Here the path-initial attribute `phn` in `<phn syl1 tail>` contrasts with the path-initial `mor` in `<mor suffix>` (see section 5). These path-initial attributes (along with `syn` and `sem`) serve to partition the feature space into phonological and morphological domains, respectively. Note also the distinction between `<tail->` and `<tail>` attributes where the former corresponds to the 'underlying' segment and the latter to its surface realization.

The `Devoice` function itself simply maps voiced stops and fricatives to their voiceless counterparts and maps all other consonants to themselves:

```
Devoice:
    <b> == p
    <d> == t
    <g> == k
    <v> == f
    <z> == s
    <> == IDEM.
```

[11] Kloeke actually argues that the term 'final consonant devoicing' is inaccurate because the feature in question is tenseness, rather than voice, but this is immaterial in an account which is formulated segmentally rather than featurally.

The definitions given earlier in this section provide the phonological skeleton for our analysis of inflection. The flesh on these bones can be filled in at any point in the hierarchy, and can be shaped by information from any domain of lexical representation, be it syntactic, morphological or phonological.[12]

4. Morphophonology

Our account of the inflection of nouns in German includes two morphophonological alternations: vowel lengthening and umlaut. Vowel lengthening occurs (along with a stress shift) in the plurals of one noun class. Umlaut occurs in the plural forms of two noun classes, sometimes accompanied by a suffix, sometimes not.

As with final consonant devoicing,[13] we describe both morphophonological alternations segmentally, even though they clearly involve feature switching. Lengthening is a change in the length feature (and/or the tense feature) whilst umlaut fronts back vowels. Although a segmental description misses the featural generalizations, our approach can be readily extended to a featural level of description, as discussed in Cahill (1993) and Gibbon (1989). Recasting the phonological and morphophonological components of our description featurally would not be difficult to do but it would make it much harder for the reader to follow the formal presentation. It would thus obscure the analysis of German noun inflection that is the main focus of the present paper.

In our account, umlaut and lengthening are no different in their roles from any other stem alternation or, indeed, from affixation. Any noun class has its inflected forms given by equations which may specify the suffix that appears, or a stem alternation, or both.

4.1 *Vowel lengthening*

Vowel lengthening occurs in the plural forms of a small set of German nouns which also exhibit a stress difference between the singular and plural roots. An example is the word *Tutor*, which has the singular form /t^u:tOr/ and

[12] Although we use different technologies, our general view on the proper relations between morphosyntax, morphotactics and morphophonology is the same as that outlined by Krieger et al. (1993: 146). The similarities and differences between the two approaches are discussed in detail in Evans & Gazdar (1996: 208–212).

[13] Wiese (1996a: 200) demonstrates with reference to the devoicing of the foreign word *Orange*/o:raNS/ that a featural definition best captures the phenomenon.

the plural /tu:t^o:r@n/. In our analysis, such nouns belong in declension class Noun_L, defined as follows:

```
Noun_L:
    <> == Noun_D
    <phn root focus sing> == syl2
    <phn syl1 peak plur> == Lengthen: <''<phn syl1 peak-
                                                plur>''>.
```

Final syllable focus (and hence stress) is inherited from Noun_D (along with the suffixes associated with declension Noun_D) via the first, empty path, equation but this is overridden for the singular root by the second equation.[14] The third equation specifies that the peak for the plural forms undergoes lengthening. The Lengthen function itself is then defined in a similar manner to final consonant devoicing in section 3.3, above:

```
Lengthen:
    <O> == o:
    <U> == u:
    <I> == i:
    <> == IDEM.
```

As can be seen, there are only three vowels which actually undergo the lengthening, /O/, /U/ and /I/. These get mapped to /o:/, /u:/ and /i:/ respectively.

4.2 *Umlaut*

German umlaut has been widely discussed and a variety of alternative accounts of it have been proposed. Historically, it is a process of vowel harmony which fronts back vowels in roots when a suffix with a front vowel is added. However, its synchronic status is somewhat different, both in its range of application and in its precise phonological realisation (Chapman 1994). In the domain of nominal inflection, it marks certain subclasses of nouns for plural, usually with an accompanying *-e* or *-er* suffix. Whilst those nouns which suffix *-er* in the plural always undergo umlaut as well, this is not true of the *-e* suffix. There is also a handful of nouns which undergo umlaut on one of their back vowels, but which have other back vowels which are unaffected. The class of nouns which undergoes umlaut cannot be determined by independent synchronic linguistic factors, be they phonological, syntactic or semantic. It is simply a matter of declensional class membership.

The realization of umlaut is more complex than a fronting of back vowels. As Wiese (1996a: 181–194, 1996b) points out, the mapping requires that the

[14] Jessen (1998, section 2.3.2.5) discusses the problem that this kind of inflectional stress shift poses for level-ordered approaches to morphology in which stress is assigned at level 1 whilst regular inflectional suffixes only appear at level 2 or 3. Declension based stress alternations of this general kind are much more common in Russian (Brown et al. 1996).

featural definitions permit /E/ to be viewed as the front variant of /a/, rather than the raised variant as one might expect from a neutral featural definition of the vowel system of German. The realisation of umlaut in the diphthongs is especially interesting. Contrary to what one might expect, it is not the case that both vowels in a diphthong are fronted when umlaut is applied. The diphthong /au/ has the umlauted version /Oy/, the second element being fronted, but the first being raised and rounded. Wiese (1996a: 182) proposes a rule of Rounding Assimilation to account for this apparent discrepancy, but this still leaves a question about the height of the vowels in question. The only other diphthong that occurs in German (with the exception of those which may appear in English loan words) is /ai/, and this is invariant. An elegant featural statement of umlaut phonology may well be possible, but it is not our concern here.

Like Trost (1990, 1991), we impose a complete separation between the phonological mapping and the morphosyntactic conditions that require the mapping to take place. Such a modular approach has the advantage that the very same phonological rule can be invoked (or not invoked) by quite different components of lexical description. Thus he notes the existence of contrasting triples such as *Hand/Hände/handlich* and *Tag/Tage/täglich*.[15] German verbs also undergo umlaut in the past tense of some strong verbs, and in the subjunctive forms of all verbs.

Reinhard (1990) provides an analysis of umlaut which resembles ours in many respects. Her representation of lexical entries is, like ours, expressed in terms of phonologically based syllable constituents, although she is describing the orthography, not the phonology. Her account, like ours, treats the realization of umlaut by means of a separate node defining the mappings from vowels to their umlauted counterparts, but she omits any reference to diphthongs. We shall discuss her definition of inflectional classes in section 5.3 below. Reinhard includes some derivational occurrences of umlaut, while we only address inflectional occurrences in the present paper.

Although umlaut occurs in both nouns and verbs in German, with identical phonological consequences, the morphosyntactic conditioning is different for the two parts of speech. For nouns we locate the statement of this conditioning at the Noun node (unsurprisingly). What we need to say can be glossed as follows: 'A peak in a root undergoes umlaut if (i) it occurs in a noun that is morphosyntactically marked for umlaut, (ii) it occurs in a plural form of the lexeme, and (iii) the peak occurs in the focussed syllable. Otherwise the peak is left unchanged'. Expressing this in DATR, we get:

[15] Trost himself deals with such phenomena monotonically by invoking an internally complex umlaut feature that contains a disjunction of feature structures where each disjunct contains a list of all the possible inflectional and derivational umlaut triggers (Trost 1993: 371–372). This analysis illustrates the rather high price that has to be paid for maintaining monotonicity in the lexicon. Dieter Wunderlich has drawn our attention to the fact that *handlich* has a slightly idiosyncratic meaning ('handy') and may thus be a frozen form.

```
Noun:
    <> == Word
    <phn $yll peak> == IF: <AND: <''<mor umlaut>''
                                 EQ: <plur>
                                 EQ: <$yll ''phn root focus>''>>
                            THEN Umlaut: <''<phn $yll peak->''>
                            ELSE         ''<phn $yll peak->''>.
```

Note the distinction between <peak-> and <peak> attributes where the former corresponds to the 'underlying' segment and the latter to its surface realization as in the exactly analogous <tail-> and <tail> attribute pair which we drew attention to in the discussion of final consonant devoicing in section 3.3 above.

The Umlaut function itself can be defined thus:[16]

```
Umlaut:
    <a>   == E
    <a:>  == E:
    <O>   == /
    <o:>  == |:
    <U>   == Y
    <u>   == Y
    <u:>  == Y:
    <a u> == O y
    <a i> == a i
    <>    == IDEM.
```

This mapping defines the alternations for all of the vowels (and the diphthong) which undergo umlaut.[17]

5. Morphology

The inflectional behaviour of German nouns is less complex than that of German verbs or adjectives and yet is significantly more complex than the noun inflection systems of many European languages. Bierwisch once remarked that 'the declension of [German] nouns is highly degenerate […] only the stem formation for singular and plural is of morphological interest' (1967: 245 n. 19). More recently, and more positively, Clahsen et al. have noted that 'the noun plural system in German is particularly interesting, because most nouns have irregular plurals in German and the regular (default) plural is less frequent than several of the irregular plurals. Thus it

[16] See Göhler (1995: 50).

[17] The observant reader will have spotted that two diphthongs appear in the list of explicit arguments to the Umlaut function. But one of them, /ai/, is simply mapped to itself. And one might reasonably have expected such a mapping to have been taken care of by the IDEM identity mapping that is invoked for all the arguments not explicitly listed. However, removal of this apparently redundant equation would lead to /ai/ getting mapped to /E/ by the first equation given. This technical infelicity could be eliminated by recoding the function. But only at the cost of obscuring how it works. We have opted for transparency.

is unclear how a language learner determines whether German even has a regular plural, and if so what form it takes' (1992: 225).

Broadly speaking, nouns are marked for number and case, with singular, plural and four cases being distinguished. However, the inflectional markers distinguish only a subset of the possibilities in the inflection of any given noun and no nouns in German differentiate more than four of the eight possibilities (e.g. *Arm* which appears as *Arm*, *Arm-e*, *Arm-s*, and *Arm-en*). Apart from two classes of nouns sometimes termed the weak nouns, which we shall discuss shortly, the nouns inflect for plural according to declensional classes, of which there are five major classes which have marked plurals and unmarked singulars (e.g. *Klub*, *Klub-s*) and six small classes which mark singular as well as plural (e.g. *Alb-um*, *Alb-en*). There is also a minor subclass which has a stem alternation other than umlaut. Whereas in the major classes the singular forms are essentially the base and citation forms of the noun in question, in the six minor classes, the base form, to which singular and plural suffixes are added can be easily identified. The citation form in these classes in the singular inflected form. The numbers of nouns in these classes is relatively small and they are largely (if not entirely) made up of words of foreign origin. However, they are sensibly treated as distinct subclasses, since their behaviour is entirely rule-governed. On the basis of the data in CELEX the smallest minor class has 5 members and the largest minor class has 182.[18]

In addition to these declensional classes which define the plural forms, there are inflections for genitive singular forms, dependent on gender, and for dative plural forms.

Our account of the inflection of German nouns is structured as a simple inheritance hierarchy which defines the affixal and morphophonological behaviour of the nouns in each declensional class by referring to general morphophonological functions as discussed above and to a network of suffix nodes described in section 5.2, below.

5.1 *Morphotactics*

As long as we restrict ourselves to matters of inflection, defining the morphotactics of German nouns is about as simple as it could be. By default, German words consist of a (focussed non-umlauting) root and a suffix and, by default, the root is a monosyllable. Given our definition of `Syllable` above this statement of German word structure can be readily encoded in DATR as follows:

```
Word:
    <> == Syllable
    <phn root focus> == syl1
    <mor umlaut> == F
    <mor word> == '' <phn root form> '' '' <mor suffix> ''.
```

[18] The total number of German nouns listed in CELEX (version 2) is 30490.

Notice that the <mor suffix> path will default to the null sequence via the link to Syllable and thence to Null unless it gets defined at a point lower in the hierarchy.

5.2 *Suffixes*

Typically, German inflectional affixes are monosyllabic. We can capture this fact by attaching the Affix node to the (mono-)Syllable node. This affix node then forms the root of a directed graph of suffix nodes. Suffixes are rather abstract entities in this analysis. The simplest suffixes are those that have a constant phonological form, independent of phonological context or morphosyntactic variation. For such suffixes, all we need to specify are the phonemes that make up their peak, body and tail, as in the nodes defined below:

```
Affix:
    <> == Syllable.

Suffix_i:
    <> == Affix
    <phn syl1 peak> == i:.

Suffix_es:
    <> == Affix
    <phn syl1 peak> == @
    <phn syl1 body> == s.

Suffix_us:
    <> == Suffix_es
    <phn syl1 peak> == U.

Suffix_is:
    <> == Suffix_es
    <phn syl1 peak> == I.
```

Other suffixes exhibit morphosyntactic variance. Thus, for example, the Suffix_um is realized as *-ums* on singular genitives and the Suffix_er is realized as *-ern* on plural datives.

```
Suffix_um:
    <> == Affix
    <phn syl1 peak> == U
    <phn syl1 body> == m
    <phn syl1 tail sing gen> == s.
```

Because morphosyntactic attributes are visible in the phonology in the framework adopted here, it is completely straightforward to capture such variance. In particular, note how both Suffix_e1 and Suffix_er inherit a component of their plural dative forms from Suffix_n (see Zwicky 1985: 383).

```
Suffix_e1:
    <> == Suffix_n
    <phn syl1 peak> == @.

Suffix_er:
    <> == Suffix_e1
    <phn syl1 body> == r.

Suffix_n:
    <> == Affix
    <phn syl1 tail plur dat> ==
                    IF:<AND:<POLYSYLLABLE:<>
                                EQ:<n ''Root:<phn syl1 tail->''>>
                    THEN Null
                    ELSE n>.
```

The definition of `Suffix_n` provides `/n/` as the tail consonant of the suffix in plural datives unless the root is a polysyllable that itself ends in `/n/`.[19]

As every undergraduate linguist eventually learns, the English *-s* suffix that is used to mark third person singular present tense on regular verbs and plural on regular nouns appears in three different phonological guises depending on the phonological context provided by the end of the root to which it is attached. Similarly, German has an *-s* suffix (`Suffix_S`, below) that requires the insertion of schwa if the root ends in a sibilant. It also has a couple of suffixes whose realization is determined by whether the final syllable of the root has a schwa peak. Zwicky suggests that the first of these `Suffix_e2`, needs to check whether the root ends 'in `schwa` plus a sonorant' (1992: 352) but the reference to sonorant appears to be redundant since every German noun in the relevant declension that has schwa as the peak of the final syllable of its root has a sonorant following schwa. The only nouns listed in CELEX as having a final syllable with a schwa followed by a non-sonorant consonant are five English loan words ending in `/@t/` (e.g. *Kricket*) which all take the *-s* plural suffix, and the archaic *Sammet*, which is given as *Samt* 'velvet' in most modern dictionaries and, as a mass noun, arguably has no plural forms in any case. In connection with the second, `Suffix_en1`, Köpcke notes that 'the schwa will be deleted in exactly those cases where the stem of a noun already ends in schwa or in schwa + consonant' (1988: 307).[20]

---

[19] We have not provided a definition of the `Root` node crucially invoked here and in the definitions of various other suffix nodes below. This is because `Root` is not a standard node name (like `Noun`, `Affix`, `Umlaut`, `Tutor`, etc.), but rather an alias for the DATR 'query node' (commonly called `Qnode`) which is essentially an indexical expression (like *here* or *now* in English) that always refers to the node at which the current query originated. Thus, for example, if one is engaged in deriving an inflected form of `Tutor`, any occurrence of `Root` will be treated exactly as if it was a reference to `Tutor`.

[20] Kloeke's statement of the rule is stress dependent (1982: 174).

```
Suffix_S:
  <> == Affix
  <phn syl1 tail> == s
  <phn syl1 peak> == IF:<SIBILANT:<''Root:<phn syl1 tail-
                     >''>
                     THEN @
                     ELSE Null>.

Suffix_e2:
  <>
  <phn syl1 rhyme> ==
                     IF:<EQ:<@ ''Root:<phn syl1 peak->''>
                     THEN Null
                     ELSE ''Suffix_e1:<phn syl1 rhyme>''>.

Suffix_en1:
  <> == Affix
  <phn syl1 body> == n
  <phn syl1 peak> == IF:<EQ:<@ ''Root:<phn syl1 peak->''>
                     THEN Null
                     ELSE @>.
```

Suffix_e2 differs from Suffix_S and Suffix_en1 in that one of its two allomorphs is the empty sequence. It would be technically possible to treat the allomorphy of Suffix_en2 in the same kind of way that Corbett & Fraser employ in their analysis of Russian genitive plurals (1993: 125; Fraser & Corbett 1995: 127). But their monadic morpheme style of analysis would not readily extend to the peak alternation found in Suffix_S and Suffix_en1. By contrast, the style of analysis we adopt here treats all three cases of phonologically conditioned allomorphy in an identical manner.

The most interesting of the German suffixes are those whose form is partly determined by an inherent morphosyntactic property of the root – specifically, the gender of the root. Thus, for example, there is a suffix (Suffix_s, below) that marks singular genitives and plurals. When it is realized at all this suffix is realized phonologically by Suffix_S considered above (thus a schwa gets inserted after sibilants) but it has no phonological realization as a singular genitive on feminine nouns.[21] There are also a couple of other suffixes that inherit this sensitivity to the gender of the root (see Carstairs-McCarthy 1992: 231).

```
Suffix_s:
  <> == Affix
  <phn syl1 peak> == <phn syl1 tail>
  <phn syl1 tail sing gen> == IF:<EQ:<femn ''Root:
                                                <syn gender>''>
                              THEN Null
                              ELSE ''Suffix_S''>
  <phn syl1 tail plur> == ''Suffix_S''.

Suffix_a:
  <> == Suffix_s
  <phn syl1 peak> == a:.
```

[21] Readers already familiar with the DATR language will note that our definition of the Suffix_s node invokes Suffix_S as a quoted node.

```
Suffix_en2:
  <> == Suffix_en1
  <phn syl1 tail> == Suffix_s.

Suffix_o:
  <> == Suffix_s
  <phn syl1 peak> == o:.
```

5.3 *Declensions*

The overall structure of our (non-monotonic) declension hierarchy is shown in Figure 1. It has the *Noun* node at the top and this is characterized by the

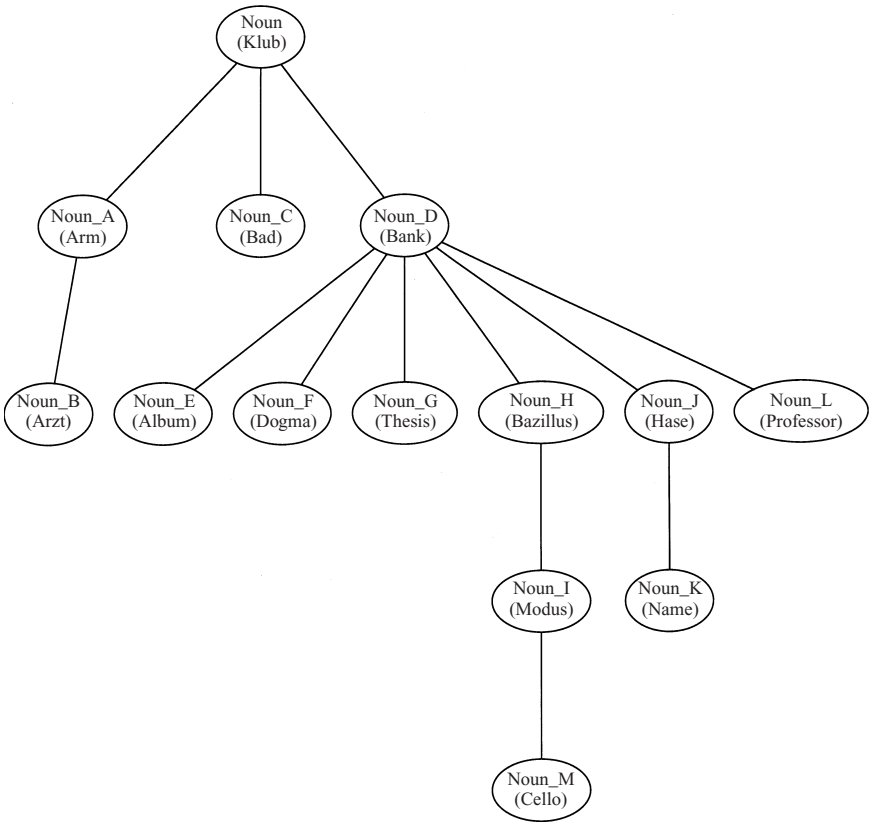

Figure 1
The noun declension hierarchy

-s suffix frequently found in genitive singulars and in the plurals of a small set of common German nouns. This latter may seem a curious choice for the default plural suffix for German since it occurs much less frequently than the

https://doi.org/10.1017/S0022226798007294 Published online by Cambridge University Press

other plural suffixes. However, we are persuaded by the extensive linguistic evidence given by Clahsen and his collaborators that *-s* is indeed the default plural suffix for German:[22] it is the suffix that standardly occurs with surnames, product names, acronyms, truncated nouns, unassimilated borrowings, foreign words, derived forms, neologisms, onomatopoeic nouns, and nouns formed from VPs and APs.

The full definition for the Noun node is shown below:[23]

```
Noun:
    <> == Word
    <syn gender> == masc
    <mor suffix> == ''Suffix_s:<phn root form>''
    <phn $yll peak> == IF:<AND:<''<mor umlaut>''
                               EQ:<plur>
                               EQ:<$yll ''<phn root focus>''>>
                       THEN Umlaut:<''<phn $yll peak->''>
                       ELSE        <phn $yll peak->''>.
```

The subclasses of nouns which follow the Noun pattern (class VIII″ for Carstairs 1987: 241) in their singulars but inflect their plural forms differently are defined as follows. Noun_A nouns (strong for Carstairs 1987: 241) have an *-e* suffix in the plural, (e.g. *Arm*, *Arm-e*); Noun_B nouns (also strong for Carstairs 1987: 241) are a subclass of Noun_A nouns, with the same suffix, but additionally an umlauted root (e.g. *Fuss*, *Füss-e*); Noun_C nouns (class VII for Carstairs 1987: 241) have an umlaut and an *-er* suffix (e.g. *Mann*, *Männ-er*); Noun_D nouns (class XIV″ for Carstairs 1987: 241) have an *-en* suffix (e.g. *Frau*, *Frau-en*). This analysis of the main noun classes is fully PEP-compliant.[24] The only substantive distinctions between his taxonomy and ours arise from our treatment of umlaut: Carstairs does not consider stem alternation to be a factor in determining paradigms.

```
Noun_A:
    <> == Noun
    <mor suffix plur> == ''Suffix_e2:<phn root form plur>''.

Noun_B:
    <> == Noun_A
    <mor umlaut> == T.
```

[22] See Clahsen et al. (1992: 228–229), Clahsen et al. (1996: 119–122), Marcus et al. (1995: 229–231). See also Carstairs-McCarthy (1992: 232), Kilbury (1995) and Wiese (1996a: 138). It is not a view that Wurzel is willing to adopt – he remarks that 'it is incompatible with the facts of German inflectional morphology!' (1990: 215 n. 6).

[23] We specify gender in our analysis only because certain suffixes are sensitive to the gender of the root. For technical convenience we make masculine the default gender and stipulate feminine and neuter on individual lexemes as appropriate. However, gender in German (and Dutch) is largely predictable on the basis of phonology, semantics and declension and, in forthcoming work, we provide a comprehensive predictive account of German gender. That account can simply replace the stipulations made in the present paper (see Fraser & Corbett (1995: 128–130) for a predictive account of Russian gender).

[24] A morphological analysis is *PEP-compliant* if and only if it is consistent with the Paradigm Economy Principle. As Carstairs notes, German nouns provide a severe challenge to that principle (1987: 234).

```
Noun_C:
    <> == Noun
    <mor umlaut> == T
    <mor suffix plur> == ''Suffix_er:<phn root form plur>''.

Noun_D:
    <> == Noun
    <mor suffix plur> == ''Suffix_en1:<phn root form plur>''.
```

The Noun_D declension itself has a variety of subclasses which inflect in different ways in their singular forms. They can be defined as follows. Noun_E nouns have an *-um* suffix in the singular (e.g. *Alb-um*, *Alb-en*); Noun_F nouns have an *-a* suffix in the singular (e.g. *Dogm-a*, *Dogm-en*); Noun_G nouns have an *-is* suffix in the singular (e.g. *Thes-is*, *Thes-en*); Noun_H nouns have an *-us* suffix in the singular (e.g. *Bazill-us*, *Bazill-en*).

```
Noun_E:
    <> == Noun_D
    <mor suffix sing> == ''Suffix_um:<phn root form sing>''.

Noun_F:
    <> == Noun_D
    <mor suffix sing> == ''Suffix_a:<phn root form sing>''.

Noun_G:
    <> == Noun_D
    <mor suffix sing> == ''Suffix_is:<phn root form sing>''.

Noun_H:
    <> == Noun_D
    <mor suffix sing> == ''Suffix_us:<phn root form sing>''.
```

Two further declensions can be defined by reference to one of those just given: Noun_I nouns have the same *-us* suffix in the singulars as Noun_H nouns but use *-i* to mark their plural forms (e.g. *Mod-us*, *Mod-i*). Noun_M nouns have the same plural suffix as Noun_I, but use *-o* to mark the singular (e.g. *Cell-o*, *Cell-i*).

```
Noun_I:
    <> == Noun_H
    <mor suffix plur> == ''Suffix_i:<phn root form plur>''.

Noun_M:
    <> == Noun_I
    <mor suffix sing> == ''Suffix_o:<phn root form sing>''.
```

It should be noted that these further classes maintain PEP-compliance since each introduces an additional suffix. Carstairs himself does not provide for most of these classes, presumably because they mostly comprise foreign words. The two foreign words which he does include, *Museum* and *Cello*, he treats as undergoing truncation plus suffixation in their plurals.[25] We make

[25] Although Carstairs treats these as individual exceptions, CELEX lists 182 nouns which behave like *Museum* and 5 which behave like *Cello*.

a distinction between the root and citation forms and provide singular suffixes. This then leads to our additional classes.

The so-called weak nouns come in two main varieties: Noun_J nouns have an -(*e*)*n* suffix for all singular and plural forms (e.g. *Hase-n*) except the nominative singular which has no suffix (e.g. *Hase*). Noun_K nouns are just like Noun_J nouns except in the singular genitive where -(*e*)*ns* appears.[26]

```
Noun_J:
    <> == Noun_D
    <mor suffix sing> == ''Suffix_en1:<phn root form sing>''.

Noun_K:
    <> == Noun_J
    <mor suffix sing gen> == ''Suffix_en2:<phn root form sing
                                                        gen>''.
```

Finally, the Noun_L declension, which is yet another subdeclension of Noun_D, has, as already noted in section 4.1 above, different focussed syllables in its singular and plural roots and, in addition, exhibits vowel lengthening in the final syllable of plurals.

```
Noun_L:
    <> == Noun_D
    <phn root focus sing> == syl2
    <phn syl1 peak plur> == Lengthen:<''<phn syl1 peak-
                                                    plur>''.
```

5.4 *Alternative declension hierarchies*

In this subsection we discuss how our noun declension hierarchy compares to other inheritance-based accounts of German noun inflection.

Reinhard (1990) gives a hierarchy of noun inflection (Figure 2)[27] but her analysis is primarily concerned with umlaut, and so does not attempt a comprehensive account of those classes where umlaut is irrelevant. Her classes E_marked, E, ER and EN broadly match our classes A, B, C and D respectively, although she subdivides A and B according to whether the *-e* suffix is realised or not. We avoid the need for this subdivision by allowing

[26] This is the one area of our account which is not fully PEP-compliant. However, as can be seen from the lengthy discussion Carstairs (1987: 241–250) himself devotes to these nouns, they are genuinely problematic for the Paradigm Economy Principle. In his view, they are undergoing a shift which will restore compliance. Kilbury (1995) demonstrates that reference to a semantic animacy feature would permit these two classes to be collapsed, with animate nouns behaving as Noun_J and inanimate nouns as Noun_K. However, *Planet* would appear to be an exception to this, being an inanimate Noun_J.

[27] In this and subsequent figures, we have given the noun class labels together with the corresponding class in our analysis. Where the classes are equivalent, this is indicated by the '=' symbol and where a class is a proper subset of one of our classes, this is indicated by the '<' symbol.

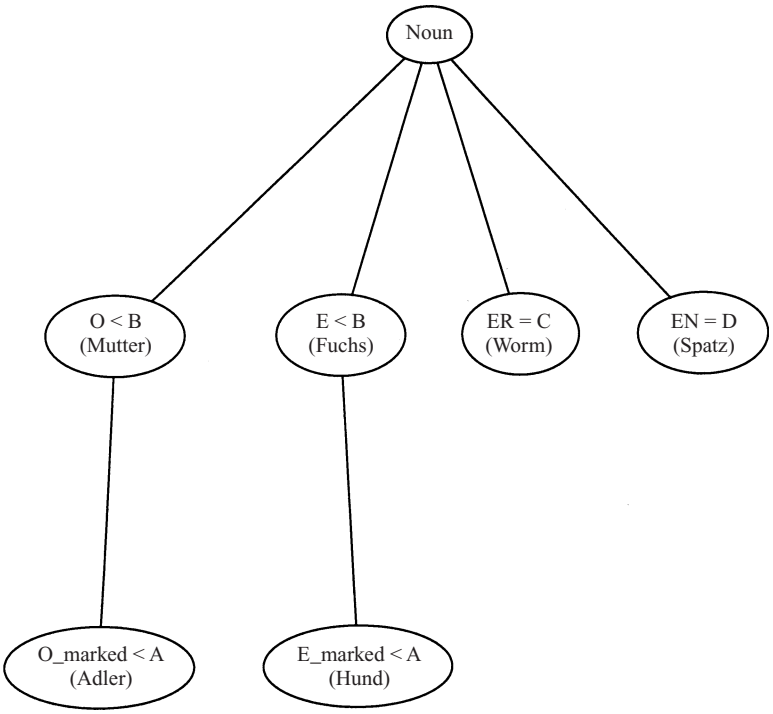

Figure 2
Reinhard's hierarchy

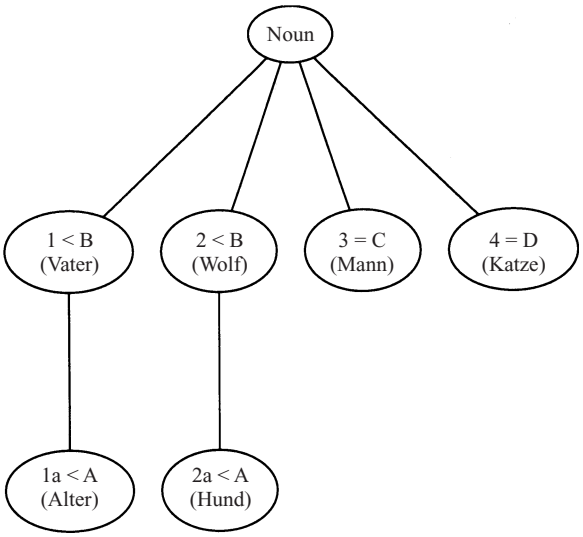

Figure 3
Gibbon's hierarchy

https://doi.org/10.1017/S0022226798007294 Published online by Cambridge University Press

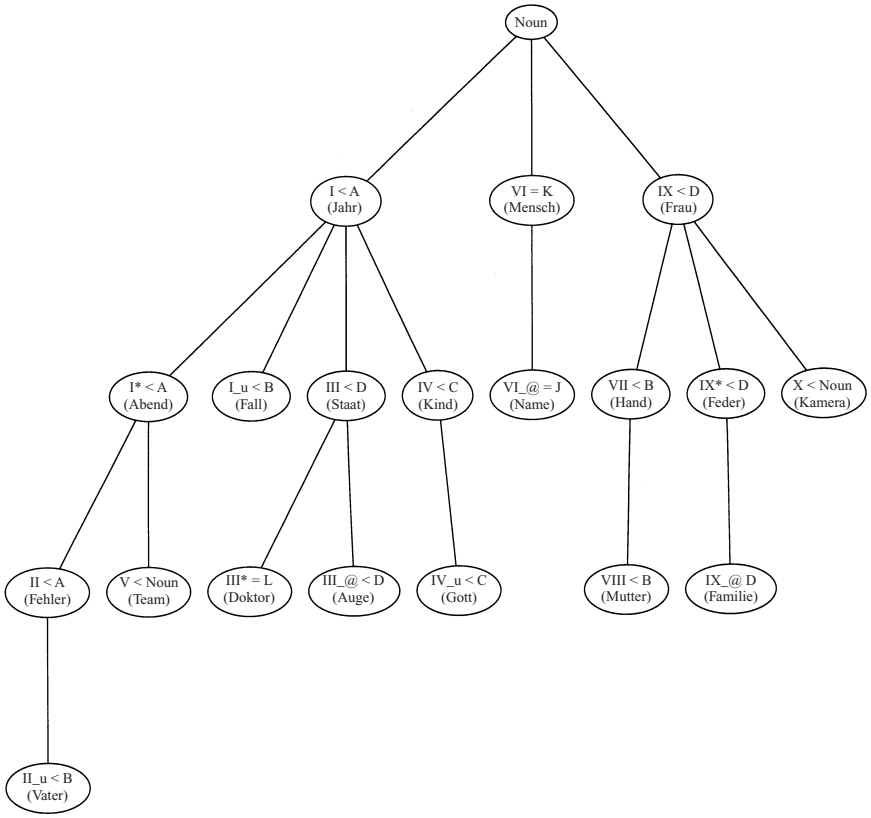

Figure 4
Bleiching's hierarchy

the realization of the suffix to depend on the phonological properties of the root. Gibbon (1992) uses the same hierarchy as Reinhard (Figure 3).

Bleiching (1994) provides an extensive hierarchy (Figure 4) which appears, at first glance, to be very different from ours. However, closer inspection reveals that most of the differences follow from the fact that we postulate suffixes whose form is determined by the gender or by phonological properties of the root. Thus, of the classes she gives, `VII`, `VIII`, `IX` and `X` are distinct from `I_u`, `II_u`, `III` and `V` respectively only in the gender (the former classes representing feminine nouns which have zero suffix in the singular genitive). Classes `II`, `II_u` and `IX*` are distinct from `I`, `I_u` and `IX` only in the presence of a final schwa syllable in the root. Class `I*` is distinct from `II` only in phonological form, being disyllabic with a non-schwa (heavy) final syllable. Classes `IV` and `IV_u` are not distinct in our account, since `IV` consists only of roots with front vowels, which are not affected by the umlaut function. Finally, the distinction between classes `III` and

https://doi.org/10.1017/S0022226798007294 Published online by Cambridge University Press

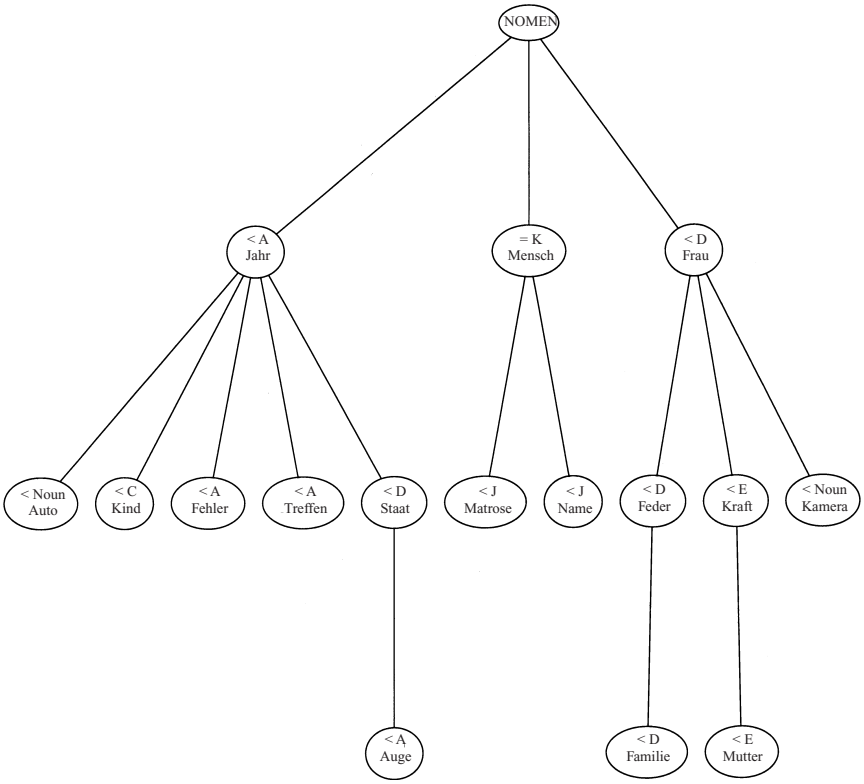

Figure 5
Bleiching et al.'s hierarchy

`III_@` and classes `IX` and `IX_@` appears to be motivated by compounding facts.

Allowing for these distinctions, her classes map to ours in the following way:

`V` → `Noun`
`I` → `A`
`I_u` → `B`
`IV` → `C`
`III` → `D`
`VI_@` → `J`
`VI` → `K`
`III*` → `L`

The main difference in the hierarchical organization of these two sets of classes stems from our assumption (defended in subsection 5.3 above) that the class that defines the *-s* plural subclass (our `Noun` class) stands at the root of the tree and defines the suffixation of genitive singulars and dative plurals.

https://doi.org/10.1017/S0022226798007294 Published online by Cambridge University Press

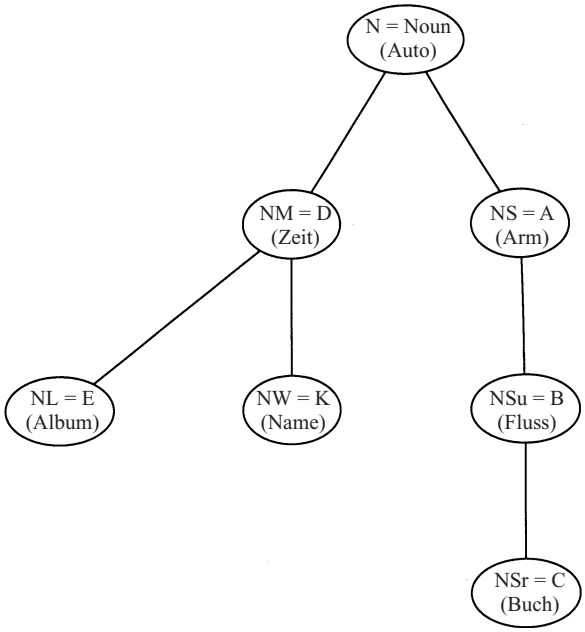

Figure 6
Kilbury's hierarchy

Bleiching et al. (1996) give a hierarchy (Figure 5) which is essentially the same as that given in Bleiching (1994), but with rather fewer subclasses.[28]

The declension hierarchy presented by Kilbury (1995) (Figure 6) comes closest to the one we propose and ours largely subsumes his. The main difference between his hierarchy and ours is that his umlaut classes inherit the umlaut and override the suffix, whereas ours inherit the suffix and override the umlaut.

All of the accounts discussed here cover our classes A, B, C and D, but none cover all of the minor subclasses which we include. Differences in the hierarchies can be accounted for largely by reference to different assumptions about the roles played by gender and phonology and by whether or not *-s* is taken as the default plural for German. We have shown that by allowing gender and phonology to determine the inflectional realisation of noun forms, we can reduce the declension classes to just those required for defining the plural alternations. As noted earlier, this yields PEP-compliance.

[28] However, two new subclasses are distinguished. Without seeing their full definitions, we are not able to determine what motivates them. One subclass has *Treffen* as an instance (which appears to differ from *Fehler* only in its gender), and the other has *Matrose* as an instance (which appears to differ from *Mensch* only in being disyllabic).

6. Lexemes

A major goal for any lexeme based account of morphology is for the individual lexical entries to say as little as possible. In the ideal case it should be possible to give the declension class and the segments that make up the root and nothing more.[29] Our lexical entry for *Mann* achieves this ideal.

```
Mann:
    <> == Noun_C
    <phn syl1 onset> == m
    <phn syl1 peak-> == a
    <phn syl1 tail-> == n.
```

From this entry and the axioms presented in the previous sections of this paper, we can derive the following theorems about *Mann*.[30]

```
Mann:
    <mor word sing nom> = m^a n
    <mor word sing acc> = m^a n
    <mor word sing gen> = m^a n s
    <mor word sing dat> = m^a n
    <mor word plur nom> = m^E n @ r
    <mor word plur acc> = m^E n @ r
    <mor word plur gen> = m^E n @ r
    <mor word plur dat> = m^E n @ r n.
```

A representative sample of lexical entries can be found in Appendix A. Those that depart from the minimalist *Mann* ideal do so for one or more of three reasons: (i) they are polysyllabic, (ii) their focal syllable is nonfinal, or (iii) they embody sui generis irregularity.

Case (i) arises because we have chosen to treat words as monosyllabic by default. If a root is polysyllabic then this is stipulated. This leads to a redundancy in the individual lexical entries: if, say, a peak is given for `syl2` then the root clearly cannot be a monosyllable. We could have avoided this redundancy with some clever DATR code that checked the lexical entry for the highest numbered defined root syllable segment (to discover how many syllables were present) and then routed the `<phn root form>` path to `Monosyllable`, `Disyllable`, `Trisyllable`, etc., as appropriate. We could have written such code but you would not have wanted to read it. From our perspective it would also have been a conceptually perverse thing to have done since it would imply that syllable structure was epiphenomenal, merely contingent upon the segments that happened to be present. We have thus chosen to live with this particular redundancy in our lexical entries.[31]

---

[29] Of course, in a featural rather than a segmental account, it would be possible to give even less information since phonotactic redundancy could be extracted and captured at higher nodes of the lexical description (see Gibbon 1989 for discussion).

[30] A complete proof of the equation for the dative plural of *Mann* is provided in Appendix B.

[31] Zwicky has pointed out that the typical morphological lexicon 'will usually admit of multiple minimizations, thanks to the fact of mutual predictability (as when it is true both

Case (ii) arises because polysyllabic roots may have nonfinal focal syllables. We have assumed that this is not systematic and must therefore be stipulated when it occurs. But, if it were the case, say, that disyllabic roots have the first syllable focussed by default then this could be captured in our analysis with some minor surgery to the way the <phn root focus> path is handled at the Word and Tri/Di/Syllable nodes.

Case (iii) just happens. Even in German. We consider some examples below.

```
Gott:
    <> == Noun_C
    <phn syl1 onset> == g
    <phn syl1 peak-> == O
    <phn syl1 tail-> == t
    <mor suffix sing gen> == ''Suffix_es: <phn root
                                          form sing gen>''.
```

If /t/ was a sibilant then *Gott* would be a (sub)regular German noun in the Noun_C declension. But it isn't and *Gott* has a pattern of inflection that is unique in the German lexicon:

```
Gott:
    <mor word sing nom> = g^O t
    <mor word sing acc> = g^O t
    <mor word sing gen> = g^O t @ s
    <mor word sing dat> = g^O t
    <mor word plur nom> = g^/ t @ r
    <mor word plur acc> = g^/ t @ r
    <mor word plur gen> = g^/ t @ r
    <mor word plur dat> = g^/ t @ r n.
```

Although the /@s/ alternant is available as a stylistic variant for most nouns, *Gott* is alone in requiring it, and is singled out as unusual in German grammar books (e.g. Hammer 1977: 20).[32]

Phosphor is also unique in that it combines the singular/plural focal syllable difference and the vowel lengthening in plurals that are the defining characteristics of Noun_L class lexemes with the plural suffixes associated with class Noun_A:

```
Phosphor:
    <> == Noun_L
    <phn root form> == Disyllable
    <phn syl2 onset> == f
    <phn syl2 peak-> == O
    <phn syl2 tail> == s
```

that nouns in a particular declension class are by default of the [MASC] gender and that nouns of the [MASC] gender are by default in that declension class)' (1992: 337). And, with specific reference to German noun inflection, Köpcke remarks that 'while gender does not predict the plural morpheme, it does limit the choice' (1988: 307).

[32] Carstairs also shows the suffix *-e* as appearing on masculine and neuter singular dative nouns. This is again a stylistic variation which is entirely optional. Hammer (1977: 20) says 'Except in certain fixed phrases, the *-e* is never essential and is in fact dying out'.

```
    <phn syll onset> == f
    <phn syll peak-> == O
    <phn syll tail-> == r
    <mor suffix> == Noun_A.

Phosphor:
    <mor word sing nom> = f^O s f O r
    <mor word sing acc> = f^O s f O r
    <mor word sing gen> = f^O s f O r s
    <mor word sing dat> = f^O s f O r
    <mor word plur nom> = f O s f^o: r @
    <mor word plur acc> = f O s f^o: r @
    <mor word plur gen> = f O s f^o: r @
    <mor word plur dat> = f O s f^o: r @ n.
```

Herz is a deviant noun, having the forms:

```
Herz: <mor word sing nom> = h * E r ts.
Herz: <mor word sing acc> = h * E r ts.
Herz: <mor word sing gen> = h * E r ts @ n s.
Herz: <mor word plur nom> = h * E r ts @ n.
Herz: <mor word plur acc> = h * E r ts @ n.
Herz: <mor word plur gen> = h * E r ts @ n.
Herz: <mor word plur dat> = h * E r ts @ n.
```

Wurzel (1990: 210) treats this as a `Noun_D`, which has degenerate forms in the genitive and dative singular. We consider it to be a `Noun_K` with an irregular accusative singular form. We choose this approach for three reasons. First, it has to define only one degenerate form as opposed to two in Wurzel's account. Secondly, it seems preferable to have the much more common uninflected form as the unexpected form, particularly as this corresponds to the forms for the default noun class in our account. Thirdly, our approach equates the nominative and accusative forms. Our account of adjectives and determiners (Cahill & Gazdar 1997) captures the fact that nominative and accusative forms are commonly linked in German, so this is a fact that fits neatly into our overall view of German inflection. We therefore give the entry for *Herz* as follows:

```
Herz:
    <> == Noun_K
    <syn gender> == neut
    <phn syll onset> == h
    <phn syll peak-> == E
    <phn syll body> == r
    <phn syll tail-> == ts
    <mor suffix sing acc> == <mor suffix sing nom>.
```

7. Conclusions

It is our hope that the analysis of German noun inflection that we have presented appears straightforward and that the definitions of the basic notions like `Syllable`, `Disyllable`, `Trisyllable`, `Stress`, `Word` and `Noun`, the declensions, and the individual lexemes (as in Appendix A)

https://doi.org/10.1017/S0022226798007294 Published online by Cambridge University Press

look to be about as simple and general as they reasonably could be given what has to be expressed. The representational complexity and particularity of our analysis, such as it is, resides in the definition of the various suffixes whose phonological realization depends on a variety of phonological, morphosyntactic and syntactic variables.

However, although our analysis is representationally both simple and general, it is inferentially quite complex. Deriving a single inflected form for one lexeme requires several hundred inferential steps in the theorem prover that interprets the DATR language that we have used for our representations.[33] An example of a full proof for a single inflected form is given in Appendix B.

REFERENCES

Baayen, Harald, Piepenbrock, Richard & van Rijn, H. (1995). The CELEX Lexical Database, Release 2 (CD-ROM). Linguistic Data Consortium, University of Pennsylvania, Philadelphia, PA.

Bierwisch, Manfred (1967). Syntactic features in morphology: general problems of so-called pronominal inflection in German. In *To honor Roman Jakobson: essays on the occasion of his seventieth birthday* (vol. 1). The Hague: Mouton. 239–270.

Bleiching, Doris (1992). Prosodisches Wissen in Lexicon. In Görz, G. (ed), *Proceedings of KONVENS-92*. Berlin: Springer-Verlag. 59–68.

Bleiching, Doris (1994). Integration von Morphophonologie und Prosodie in ein hierarchisches Lexicon. In Trost, Harald (ed.), *Proceedings of KONVENS-94*. Vienna: Österreichische Gesellschaft für Artificial Intelligence. 32–41.

Bleiching, Doris, Drexel, Guido & Gibbon, Dafydd (1996). Ein Synkretismusmodell für die deutsche Morphologie. In Gibbon, Dafydd (ed.), *Natural language processing and speech technology: results of the 3rd KONVENS Conference, Bielefeld*. Berlin: Mouton de Gruyten. 237–248.

Brown, Dunstan, Corbett, Greville, Fraser, Norman, Hippisley, Andrew & Timberlake, Alan (1996). Russian noun stress and network morphology. *Linguistics* **34.1**. 53–107.

Brown, Dunstan & Hippisley, Andrew (1994). Conflict in Russian genitive plural assignment: a solution represented in DATR. *Journal of Slavic Linguistics* **2.1**. 48–76.

Cahill, Lynne (1990a). Syllable-based morphology. *Proceedings of the 13th International Conference on Computational Linguistics, COLING-90* (vol. 3). 48–53.

Cahill, Lynne (1990b). *Syllable-based morphology for natural language processing*. D.Phil. dissertation, University of Sussex. [Available as *Cognitive Studies Research Paper* **181**.]

Cahill, Lynne (1993). Morphonology in the lexicon. *Sixth Conference of the European Chapter of the Association for Computational Linguistics*. 87–96.

Cahill, Lynne & Gazdar, Gerald (1997). The inflectional phonology of German adjectives, determiners and pronouns. *Linguistics* **35.2**. 211–245.

Carstairs[-McCarthy], Andrew (1987). *Allomorphy in inflection*. London: Croom Helm.

Carstairs-McCarthy, Andrew (1992). *Current morphology*. London: Routledge.

Chapman, Carol (1994). A diachronic argument against the split morphology hypothesis – analogical umlaut in German dialects. *Transactions of the Philological Society* **92.1**. 25–39.

Clahsen, Harald, Rothweiler, Monika, Wöst, Andreas & Marcus, Gary F. (1992). Regular and irregular inflection in the acquisition of German noun plurals. *Cognition* **45**. 225–255.

Clahsen, Harald, Marcus, Gary F., Bartke, Susanne & Wiese, Richard (1996). Compounding and inflection in German child language. Booij, Geert & van Marle, Jaap (eds.), *Yearbook of Morphology 1995*. Dordrecht: Kluwer. 115–142.

Corbett, Greville & Fraser, Norman (1993). Network Morphology: a DATR account of Russian nominal inflection. *Journal of Linguistics* **29**. 113–142.

[33] See Keller (1996) for a formal specification of DATR inference.

Evans, Roger & Gazdar, Gerald (1996). DATR: a language for lexical knowledge representation. *Computational Linguistics* **22.2**. 167–216.
Fraser, Norman & Corbett, Greville (1995). Gender, animacy, and declensional class assignment: a unified account for Russian. In Booij, Geert & van Marle, Jaap (eds.), *Yearbook of Morphology 1994*. Dordrecht: Kluwer. 123–150.
Fraser, Norman & Corbett, Greville (1997). Defaults in Arapesh. *Lingua* **103**. 25–57.
Gibbon, Dafydd (1989). `phonlex1.dtr` – underspecification in phonology. Unpublished DATR fragment, University of Bielefeld.
Gibbon, Dafydd (1990). Prosodic association by template inheritance. In Daelemans, Walter & Gazdar, Gerald (eds.), *Proceedings of the Workshop on Inheritance in Natural Language Processing*. Tilburg: Institute for Language Technology. 65–81.
Gibbon, Dafydd (1992). ILEX: a linguistic approach to computational lexica. In Klenk, Ursula (ed.), *Computatio Linguae: Aufsatze zur algorithmischen und quantitativen Analyse der Sprache* (*Zeitschrift fur Dialektologie und Linguistik*, Beiheft **73**). Stuttgart: Franz Steiner Verlag. 32–53.
Gibbon, Dafydd & Bleiching, Doris (1991). An ILEX model for German compound stress in DATR. *Proceedings of the FORWISS-ASL Workshop on Prosody in Man-Machine Communication*. 1–6.
Göhler, Andreas (1995). Ein auf Default-Vererbung basierender Ansatz zur Darstellung derivationeller Morphologie im Lexikon. Magisterarbeit, Technische Universität Berlin.
Hammer, A. E. (1977). *German grammar and usage*. London: Edward Arnold.
Jessen, Michael (1998). German. In van der Hulst, Harry (ed.), *Word prosodic systems in the languages of Europe*. Berlin & New York: Mouton de Gruyter.
Keller, Bill (1995). DATR theories and DATR models. *Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics*. 55–62.
Keller, Bill (1996). An evaluation semantics for DATR theories. *Proceedings of the 16th International Conference on Computational Linguistics, COLING-96*. Copenhagen. 646–651.
Kilbury, James (1995). German noun inflection. Paper presented to the 3rd ESRC Workshop *Frontiers of Research in Morphology*, Brighton, June 1995.
van Lessen Kloeke, Wus (1982). *Deutsche Phonologie und Morphologie: Merkmale und Markiertheit*. Tübingen: Max Niemeyer Verlag.
Köpcke, Klaus-Michael (1988). Schemas in German plural formation. *Lingua* **74**. 303–335.
Krieger, Hans-Ulrich, Pirker, Hannes & Nerbonne, John (1993). Feature-based allomorphy. *Proceedings of the 31st Annual Meeting of the Association for Computational Linguistics*. 140–147.
Lieber, Rochelle (1981). *On the organization of the lexicon*. Bloomington: Indiana University Linguistics Club. [Ph.D. dissertation, MIT.]
Marcus, Gary F., Brinkmann, Ursula, Clahsen, Harald, Wiese, Richard & Pinker, Steven (1995). German inflection: the exception that proves the rule. *Cognitive Psychology* **29**. 189–256.
van Marle, Jaap (1985). *On the paradigmatic dimension of morphological creativity*. Dordrecht: Foris.
Matthews, Peter (1972). *Inflectional morphology: a theoretical study based on aspects of Latin verb conjugation*. Cambridge: Cambridge University Press.
Reinhard, Sabine (1990). Verarbeitungsprobleme nichtlinearer Morphologien: Umlautbeschreibung in einem hierarchischen Lexikon. In Rieger, Burghard & Schäder, Burkhard (eds.), *Lexikon und Lexikographie*. Hildesheim: Olms Verlag. 45–61.
Reinhard, Sabine & Gibbon, Dafydd (1991). Prosodic inheritance and morphological generalisations. *Fifth Conference of the European Chapter of the Association for Computational Linguistics*. 131–136.
Shieber, Stuart M. (1986). *An introduction to unification approaches to grammar*. Stanford: CSLI/Chicago University Press.
Stump, Gregory T. (1992). On the theoretical status of position class restrictions on inflectional affixes. In Booij, Geert & van Marle, Jaap (eds.), *Yearbook of Morphology 1991*. Dordrecht: Kluwer. 211–241.
Stump, Gregory T. (1993a). On rules of referral. *Language* **69**. 449–479.
Stump, Gregory T. (1993b). Reconstituting morphology: the case of Bantu preprefixation. *Linguistic Analysis* **23**. 169–204.

Stump, Gregory T. (1993c). Position classes and morphological theory. In Booij, Geert & van Marle, Jaap (eds.), *Yearbook of Morphology 1992*. Dordrecht: Kluwer. 129–180.
Stump, Gregory T. (1995). The uniformity of head marking in inflectional morphology. In Booij, Geert & van Marle, Jaap (eds.), *Yearbook of Morphology 1994*. Dordrecht: Kluwer. 245–296.
Trost, Harald (1990). The application of two-level morphology to non-concatenative German morphology. *Proceedings of the 13th International Conference on Computational Linguistics, COLING-90* (vol. 2). 371–376.
Trost, Harald (1991). X2MORF: a morphological component based on augmented two-level morphology. *Proceedings of the 12th International Joint Conference on Artificial Intelligence* (vol. 2). 1024–1030.
Trost, Harald (1993). Coping with derivation in a morphological component. *Sixth Conference of the European Chapter of the Association for Computational Linguistics*. 368–376.
Wells, John C. (1987). Computer coded phonetic transcription. *Journal of the International Phonetic Association* **17**. 94–114.
Wiese, Richard (1996a). *The phonology of German*. Oxford: Clarendon Press.
Wiese, Richard (1996b). Phonological versus morphological rules: on German Umlaut and Ablaut. *Journal of Linguistics* **32**. 113–135.
Wurzel, Wolfgang (1970). *Studien zur deutschen Lautstruktur*. Berlin: Akademie Verlag.
Wurzel, Wolfgang (1984). *Flexionsmorphologie und Natürlichkeit. Ein Beitrag zur morphologischen Theoriebildung*. Berlin: Akademie Verlag. [Published in English translation as *Inflectional Morphology and Naturalness*. Dordrecht: Kluwer, 1989.]
Wurzel, Wolfgang (1990). The mechanism of inflection: lexicon representations, rules, and irregularities. In Dressler, Wolfgang U., Luschützky, Hans C., Pfeiffer, Oskar E. & Rennison, John R. (eds.), *Contemporary morphology*. Berlin: Mouton de Gruyter. 203–216.
Zwicky, Arnold (1985). How to describe inflection. *Berkeley Linguistic Society* **11**. 371–386.
Zwicky, Arnold (1990). Inflectional morphology as a (sub)component of grammar. In Dressler, Wolfgang U., Luschützky, Hans C., Pfeiffer, Oskar E. & Renninson, John R (eds.), *Contemporary morphology*. Berlin: Mouton de Gruyter. 217–236.
Zwicky, Arnold (1992). Some choices in the theory of morphology. In Levine, Robert D. (ed.), *Formal grammar: theory and implementation*. Oxford: Oxford University Press. 327–371.

Authors' address: School of Cognitive & Computing Sciences,
University of Sussex,
Brighton BN1 9QH,
U.K.
E-mail: Lynne.Cahill/Gerald.Gazdar@cogs.sussex.ac.uk

APPENDIX A

A representative list of example lexical entries

```
Adler:
  < > = = Noun_A
  <phn root form> = = Disyllable
  <phn root focus> = = syl2
  <phn syl2 peak-> = = a:
  <phn syl2 tail> = = d
  <phn syl1 onset> = = l
  <phn syl1 peak-> = = @
  <phn syl1 tail-> = = r.

Album:
  < > = = Noun_E
  <syn gender> = = neut
  <phn syl1 peak-> = = a
  <phn syl1 body> = = l
  <phn syl1 tail-> = = b.

Arm:
  < > = = Noun_A
  <phn syl1 peak-> = = a
  <phn syl1 body> = = r
  <phn syl1 tail-> = = m.

Arzt:
  < > = = Noun_B
  <phn syl1 peak-> = = a:
  <phn syl1 body> = = r t
  <phn syl1 tail-> = = s t.

Bad:
  < > = = Noun_C
  <syn gender> = = neut
  <phn syl1 onset> = = b
  <phn syl1 peak-> = = a:
  <phn syl1 tail-> = = d.


Bank:
  < > = = Noun_D
  <syn gender> = = femn
  <phn syl1 onset> = = b
  <phn syl1 peak-> = = a
  <phn syl1 body> = = n
  <phn syl1 tail-> = = k.

Bazillus:
  < > = = Noun_H
  <phn root form> = = Disyllable
  <phn syl2 onset> = = b
  <phn syl2 peak-> = = a
  <phn syl1 onset> = = ts
  <phn syl1 peak-> = = I
  <phn syl1 tail-> = = l.

Blatt:
  < > = = Noun_C
  <syn gender> = = neut
  <phn syl1 onset> = = b l
  <phn syl1 peak-> = = a
  <phn syl1 tail-> = = t.

Braut:
  < > = = Noun_B
  <syn gender> = = femn
  <phn syl1 onset> = = b r
  <phn syl1 peak-> = = a u
  <phn syl1 tail-> = = t.

Cello:
  < > = = Noun_M
  <syn gender> = = neut
  <phn syl1 onset> = = t S
  <phn syl1 peak-> = = E
  <phn syl1 tail-> = = l.

Dogma:
  < > = = Noun_F
  <syn gender> = = neut
  <phn syl1 onset> = = d
  <phn syl1 peak-> = = O
  <phn syl1 body> = = g
  <phn syl1 tail-> = = m.

Fels:
  < > = = Noun_K
  <phn syl1 onset> = = f
  <phn syl1 peak-> = = E
  <phn syl1 body> = = l
  <phn syl1 tail-> = = s.

Firma:
  < > = = Noun_F
  <syn gender> = = femn
  <phn syl1 onset> = = f
  <phn syl1 peak-> = = I
  <phn syl1 body> = = r
  <phn syl1 tail-> = = m.

Fisch:
  < > = = Noun_A
  <phn syl1 onset> = = f
  <phn syl1 peak-> = = I
  <phn syl1 tail-> = = S.

Frau:
  < > = = Noun_D
  <syn gender> = = femn
```


https://doi.org/10.1017/S0022226798007294 Published online by Cambridge University Press

```
  <phn syl1 onset> = = f r
  <phn syl1 peak-> = = a u.

Fuss:
  < > = = Noun_B
  <phn syl1 onset> = = f
  <phn syl1 peak-> = = u:
  <phn syl1 tail-> = = s.

Glied:
  < > = = Noun_C
  <syn gender> = = neut
  <phn syl1 onset> = = g l
  <phn syl1 peak-> = = i:
  <phn syl1 tail-> = = d.

Gott:
  < > = = Noun_C
  <phn syl1 onset> = = g
  <phn syl1 peak-> = = O
  <phn syl1 tail-> = = t
  <mor suffix sing gen> = =
  ''Suffix_es:<phn root form
  sing gen>''.

Gruen:
  < > = = Noun_A
  <syn gender> = = neut
  <phn syl1 onset> = = g r
  <phn syl1 peak-> = = Y:
  <phn syl1 tail-> = = n.

Hase:
  < > = = Noun_J
  <phn root form> = = Disyllable
  <phn root focus sing> = = syl2
  <phn syl1 onset> = = h
  <phn syl1 peak-> = = a:
  <phn syl2 tail> = = z
  <phn syl1 peak-> = = @.

Herz:
  < > = = Noun_K
  <syn gender> = = neut
  <phn syl1 onset> = = h
  <phn syl1 peak-> = = E
  <phn syl1 body> = = r
  <phn syl1 tail-> = = ts
  <mor suffix sing acc> = =
  <mor suffix sing nom>.
Hund:
  < > = = Noun_A
  <phn syl1 onset> = = h
  <phn syl1 peak-> = = U
  <phn syl1 body> = = n
  <phn syl1 tail-> = = d.

Industrie:
  < > = = Noun_D
  <syn gender> = = femn
  <phn root form> = =
  Trisyllable
  <phn syl3 peak-> = = I
  <phn syl3 tail> = = n
  <phn syl2 onset> = = d
  <phn syl2 peak-> = = U
  <phn syl2 tail> = = s
  <phn syl1 onset> = = t r
  <phn syl1 peak-> = = i:.

Kaktus:
  < > = = Noun_H
  <phn root form> = = Disyllable
  <phn root focus sing> = = syl2
  <phn root focus plur> = = syl1
  <phn syl2 onset> = = k
  <phn syl2 peak-> = = a
  <phn syl2 body> = = k
  <phn syl2 tail> = = t
  <phn syl1 peak- plur> = = e:.

Klub:
  < > = = Noun
  <phn syl1 onset> = = k l
  <phn syl1 peak-> = = U
  <phn syl1 tail-> = = b.

Knopf:
  < > = = Noun_B
  <phn syl1 onset> = = k n
  <phn syl1 peak-> = = O
  <phn syl1 tail-> = = pf.

Laden:
  < > = = Noun_B
  <phn root form> = = Disyllable
  <phn root focus> = = syl2
  <phn syl2 onset> = = l
  <phn syl2 peak-> = = a:
  <phn syl1 onset> = = d
  <phn syl1 peak-> = = @
  <phn syl1 tail-> = = n.

Mann:
  < > = = Noun_C
  <phn syl1 onset> = = m
  <phn syl1 peak-> = = a
  <phn syl1 tail-> = = n.

Medikus:
  < > = = Noun_I
  <phn root form> = = Disyllable
  <phn root focus> = = syl2
  <phn syl2 onset> = = m
  <phn syl2 peak-> = = e:
  <phn syl1 onset> = = d
  <phn syl1 peak-> = = i:
```


https://doi.org/10.1017/S0022226798007294 Published online by Cambridge University Press

```
  <phn syl1 tail-> == k
  <phn syl1 coda plur> == ts.

Modus:
  < > == Noun_I
  <phn syl1 onset> == m
  <phn syl1 peak-> == o:
  <phn syl1 tail-> == d.


Nacht:
  < > == Noun_B
  <syn gender> == femn
  <phn syl1 onset> == n
  <phn syl1 peak-> == a
  <phn syl1 body> == x
  <phn syl1 tail-> == t.

Name:
  < > == Noun_K
  <phn syl1 onset> == n
  <phn syl1 peak-> == a:
  <phn syl1 tail-> == m.

Obst:
  < > == Noun_A
  <syn gender> == neut
  <phn syl1 peak-> == o:
  <phn syl1 body> == p s
  <phn syl1 tail-> == t
  <mor word plur> == UNDEF.

Ochse:
  < > == Noun_J
  <phn root form> == Disyllable
  <phn root focus> == syl2
  <phn syl2 peak-> == O
  <phn syl2 tail> == k
  <phn syl1 onset> == s
  <phn syl1 peak-> == @
  <phn syl1 tail-> == Null.

Orgel:
  < > == Noun_D
  <syn gender> == femn
  <phn root form> == Disyllable
  <phn root focus> == syl2
  <phn syl2 peak-> == O
  <phn syl2 tail> == r
  <phn syl1 onset> == g
  <phn syl1peak-> == @
  <phn syl1 tail-> == l.

Phosphor:
  < > == Noun_L
  <phn root form> == Disyllable
  <phn syl2 onset> == f
  <phn syl2 peak-> == O
  <phn syl2 tail> == s
  <phn syl1 onset> == f
  <phn syl1 peak-> == O
  <phn syl1 tail-> == r
  <mor suffix> == Noun_A.

Planet:
  < > == Noun_J
  <phn root form> == Disyllable
  <phn syl2 onset> == p l
  <phn syl2 peak-> == a
  <phn syl1 onset> == n
  <phn syl1 peak-> == e:
  <phn syl1 tail-> == t.

Professor:
  < > == Noun_L
  <phn root form> == syllable
  <phn syl3 onset> == p r
  <phn syl3 peak-> == o:
  <phn syl2 onset> == f
  <phn syl2 peak-> == E
  <phn syl1 onset> == s
  <phn syl1 peak-> == O
  <phn syl1 tail-> == r.

Riff:
  < > == Noun_A
  <phn syl1 onset> == r
  <phn syl1 peak-> == I
  <phn syl1 tail-> == f.

Schatten:
  < > == Noun_A
  <phn root form> == Disyllable
  <phn root focus> == syl2
  <phn syl2 onset> == S
  <phn syl2 peak-> == a
  <phn syl1 onset> == t
  <phn syl1 peak-> == @
  <phn syl1 tail-> == n.

Schiff:
  < > == Noun_A
  <syn gender> == neut
  <phn syl1 onset> == S
  <phn syl1 peak-> == I
  <phn syl1 tail-> == f.

Schmied:
  < > == Noun_A
  <phn syl1 onset> == S m
  <phn syl1 peak-> == i:
  <phn syl1 tail-> == d.

Sohn:
  < > == Noun_B
  <phn syl1 onset> == z
  <phn syl1 peak-> == o:
  <phn syl1 tail-> == n.
```


https://doi.org/10.1017/S0022226798007294 Published online by Cambridge University Press

```
Sonne:
  < > = = Noun_D
  <syn gender> = = femn
  <phn root form> = = Disyllable
  <phn root focus> = = syl2
  <phn syl2 onset> = = z
  <phn syl2 peak-> = = O
  <phn syl1 onset> = = n
  <phn syl1 peak-> = = @

Staat:
  < > = = Noun_D
  <phn syl1 onset> = = S t
  <phn syl1 peak-> = = a:
  <phn syl1 tail-> = = t.

Stadt:
  < > = = Noun_B
  <syn gender> = = femn
  <phn syl1 onset> = = S t
  <phn syl1 peak-> = = a
  <phn syl1 tail-> = = t.

Thesis:
  < > = = Noun_G
  <syn gender> = = femn
  <phn syl1 onset> = = t
  <phn syl1 peak-> = = e:
  <phn syl1 tail-> = = z.

Tuer:
  < > = = Noun_D
  <syn gender> = = femn
  <phn syl1 onset> = = t
  <phn syl1 peak-> = = y:
  <phn syl1 tail-> = = r.

Tutor:
  < > = = Noun_L
  <phn root form> = = Disyllable
  <phn syl2 onset> = = t
  <phn syl2 peak-> = = u:
  <phn syl1 onset> = = t
  <phn syl1 peak-> = = O
  <phn syl1 tail-> = = r.

Wort1:
  < > = = Noun_A
  <syn gender> = = neut
  <phn syl1 onset> = = v
  <phn syl1 peak-> = = O
  <phn syl1 body> = = r
  <phn syl1 tail-> = = t.

Wort2:
  < > = = Noun_C
  <syn gender> = = neut
  <phn syl1 onset> = = v
  <phn syl1 peak-> = = O
  <phn syl1 body> = = r
  <phn syl1 tail-> = = t.

Zeitung:
  < > = = Noun_D
  <syn gender> = = femn
  <phn root form> = = Disyllable
  <phn root focus> = = syl2
  <phn syl2 onset> = = ts
  <phn syl2 peak-> = = a i
  <phn syl1 onset> = = t
  <phn syl1 peak-> = = U
  <phn syl1 tail- = = N.
```


https://doi.org/10.1017/S0022226798007294 Published online by Cambridge University Press

APPENDIX B

An example proof

This appendix shows (in a reduced form) the full proof that the dative plural of *Mann* is /m^En@rn/ given the axioms presented in this paper. Unindented lines show the local environment at that point in the proof and the indented lines show the applicable axiom that leads to the next local environment. The sequence of seven phonemes that constitutes the value of Mann:<mor word plur dat> 'emerges' in the course of the proof and each has been explicitly annotated from [1 = m] to [7 = n] to make it easier to track their appearance which is sometimes obscured by their realization via the $char variable.

```
Mann:<mor word plur dat>
|   Man:<> == Noun_C
Noun_C:<mor word plur dat>
|   Noun_C:<> == Noun
Noun:<mor word plur dat>
|   Noun:<> == Word
Word:<mor word plur dat>
|   Word:<mor word> == ''<phn root form>'' ''<mor suffix>''
Mann:<phn root form plur dat>
|   Mann:<> == Noun_C
Noun_C:<phn root form plur dat>
|   Noun_C:<> == Noun
Noun:<phn root form plur dat>
|   Noun:<> == Word
Word:<phn root form plur dat>
|   Word:<> == Syllable
Syllable:<phn root form plur dat>
|   Syllable:<phn root> == <phn syl1>
Syllable:<phn syl1 form plur dat>
|   Syllable:<phn $yll form> == ''<phn $yll onset>''
|                               ''<phn $yll rhyme>''
Mann:<phn syl1 onset plur dat>
|   Mann:<phn syl1 onset> == m                              [1 = m]
Mann:<phn syl1 rhyme plur dat>
|   Mann:<> == Noun_C
Noun_C:<phn syl1 rhyme plur dat>
|   Noun_C:<> == Noun
Noun:<phn syl1 rhyme plur dat>
|   Noun:<> == Word
Word:<phn syl1 rhyme plur dat>
|   Word:<> == Syllable
Syllable:<phn syl1 rhyme plur dat>
|   Syllable:<phy $yll rhyme> == Stress ''<phn $yll peak>''
|                                       ''<phn $yll coda>''
Stress:<phn syl1 rhyme plur dat>
|   Stress:<phn $yll rhyme> == IF:<EQ:<$yll ''<phn root
    focus>''>
|                             THEN ^
|                             ELSE Null>
Mann:<phn root focus plur dat>
```


https://doi.org/10.1017/S0022226798007294 Published online by Cambridge University Press

```
|   Mann:<> == Noun_C
Noun_C:<phn root focus plur dat>
|   Noun_C:<> == Noun
Noun:<phn root focus plur dat>
|   Noun:<> == Word
Word:<phn root focus plur dat>
|   Word:<phn root focus> == syl1
EQ:<syl1 syl1 plur dat>
|   EQ:<$x $x> == T
Null:<phn syl1 rhyme plur dat>
|   Null:<> ==
IF:<T THEN^ELSE plur dat>
|   IF:<T THEN> == IDEM:<>
IDEM:<^ELSE plur dat>
|   IDEM:<$char> == $char <>                                  [2 = ^]
IDEM:<ELSE plur dat>
|   IDEM:<> == Null
Null:<ELSE plur dat>
|   Null:<> ==
Mann:<phn syl1 peak plur dat>
|   Mann:<> == Noun_C
Noun_C:<phn syl1 peak plur dat>
|   Noun_C:<> == Noun
Noun:<phn syl1 peak plur dat>
|   Noun:<phn $yll peak> == IF:<AND:<''<mor umlaut>''
|                                EQ:<plur>
|                                EQ:<$yll ''phn root
                                                  focus>''>>
|                          THEN Umlaut:<''<phn $yll peak-
                                                       >''>
|                                ELSE ''<phn $yll peak->''>
Mann:<mor umlaut plur dat>
|   Mann:<> == Noun_C
Noun_C:<mor umlaut plur dat>
|   Noun_C:<mor umlaut> == T
EQ:<plur plur dat>
|   EQ:<$x $x> == T
Mann:<phn root focus plur dat>
|   Mann:<> == Noun_C
Noun_C:<phn root focus plur dat>
|   Noun_C:<> == Noun
Noun:<phn root focus plur dat>
|   Noun:<> == Word
Word:<phn root focus plur dat>
|   Word:<phn root focus> == syl1
EQ:<syl1 syl1 plur dat>
|   EQ:<$x $x> == T
AND:<T T T plur dat>
|   AND:<T> == <>
AND:<T T plur dat>
|   AND:<T> == <>
AND:<T plur dat>
|   AND:<T> == <>
AND:<plur dat>
|   AND:<> == T
Mann:<phn syl1 peak- plur dat>
|   Mann:<phn syl1 peak-> == a
Umlaut:<a plur dat>
```


https://doi.org/10.1017/S0022226798007294 Published online by Cambridge University Press

```
|   Umlaut:<a> == E
Mann:<phn syl1 peak- plur dat>
|   Mann:<phn syl1 peak-> == a
IF:<T THEN E ELSE a plur dat>
|   IF:<T THEN> == IDEM:< >
IDEM:<E ELSE a plur dat>
|   IDEM:<$char> == $char < >                                  [3 = E]
IDEM:<ELSE a plur dat>
|   IDEM:< > == Null
Null:<ELSE a plur dat>
|   Null:< > ==
Mann:<phn syl1 coda plur dat>
|   Mann:< > == Noun_C
Noun_C:<phn syl1 coda plur dat>
|   Noun_C:< > == Noun
Noun:<phn syl1 coda plur dat>
|   Noun:< > == Word
Word:<phn syl1 coda plur dat>
|   Word:< > == Syllable
Syllable:<phn syl1 coda plur dat>
|   Syllable:<phn $yll coda> == ''<phn $yll body>''
|                                ''<phn $yll tail>''
Mann:<phn syl1 body plur dat>
|   Mann:< > == Noun_C
Noun_C:<phn syl1 body plur dat>
|   Noun_C:< > == Noun
Noun:<phn syl1 body plur dat>
|   Noun:< > == Word
Word:<phn syl1 body plur dat>
|   Word:< > == Syllable
Syllable:<phn syl1 body plur dat>
|   Syllable:< > == Null
Null:<phn syl1 body plur dat>
|   Null:< > ==
Mann:<phn syl1 tail plur dat>
|   Mann:< > == Noun_C
Noun_C:<phn syl1 tail plur dat>
|   Noun_C:< > == Noun
Noun:<phn syl1 tail plur dat>
|   Noun:< > == Word
Word:<phn syl1 tail plur dat>
|   Word:<phn syl1 tail> == IF: <VOWEL:<''<mor suffix>''>
|                          THEN ''<phn syl1 tail->''
|   ELSE Devoice:<''<phn syl1 tail->''>>
Mann:<mor suffix plur dat>
|   Mann:< > == Noun_C
Noun_C:<mor suffix plur dat>
|   Noun_C:<mor suffix plur> ==
|                              ''Suffix_er:<phn root form
                                                    plur>''
Suffix_er:<phn root form plur dat>
|   Suffix_er:< > == Suffix_e1
Suffix_e1:<phn root form plur dat>
|   Suffix_e1:< > == Suffix_n
Suffix_n:<phn root form plur dat>
|   Suffix_n:< > == Affix
Affix:<phn root form plur dat>
|   Affix:< > == Syllable
```


https://doi.org/10.1017/S0022226798007294 Published online by Cambridge University Press

```
Syllable:<phn root form plur dat>
|   Syllable:<phn root> == <phn syl1>
Syllable:<phn syl1 form plur dat>
|   Syllable:<phn $yll form> == ''<phn $yll onset>''
|                                ''<phn $yll rhyme>''
Suffix_er:<phn syl1 onset plur dat>
|   Suffix_er:< > == Suffix_e1
Suffix_e1:<phn syl1 onset plur dat>
|   Suffix_e1:< > == Suffix_n
Suffix_n:<phn syl1 onset plur dat>
|   Suffix_n:< > == Affix
Affix:<phn syl1 onset plur dat>
|   Affix:< > == Syllable
Syllable:<phn syl1 onset plur dat>
|   Syllable:< > == Null
Null:<phn syl1 onset plur dat>
|   Null:< > ==
Suffix_er:<phn syl1 rhyme plur dat>
|   Suffix_er:< > == Suffix_e1
Suffix_e1:<phn syl1 rhyme plur dat>
|   Suffix_e1:< > == Suffix_n
Suffix_n:<phn syl1 rhyme plur dat>
|   Suffix_n:< > == Affix
Affix:<phn syl1 rhyme plur dat>
|   Affix:< > == Syllable
Syllable:<phn syl1 rhyme plur dat>
|   Syllable:<phn $yll rhyme> == Stress ''<phn $yll peak>''
|                                      ''<phn $yll coda>''
Stress:<phn syl1 rhyme plur dat>
|   Stress:<phn $yll rhyme> == IF:<EQ:<$yll ''<phn root
                                                    focus>''>
|                              THEN ^
|                              ELSE Null>
Suffix_er:<phn root focus plur dat>
|   Suffix_er:< > == Suffix_e1
Suffix_e1:<phn root focus plur dat>
|   Suffix_e1:< > == Suffix_n
Suffix_n:<phn root focus plur dat>
|   Suffix_n:< > == Affix
Affix:<phn root focus plur dat>
|   Affix:< > == Syllable
Syllable:<phn root focus plur dat>
|   Syllable:<phn root> == <phn syl1>
Syllable:<phn syl1 focus plur dat>
|   Syllable:< > == Null
Null:<phn syl1 focus plur dat>
|   Null:< > ==
EQ:<syl1 plur dat>
|   EQ:< > == F
Null:<phn syl1 rhyme plur dat>
|   Null:< > ==
IF:<F THEN ^ ELSE plur dat>
|   IF:<F THEN $char> == <F THEN>
IF:<F THEN ELSE plur dat>
|   IF:<F THEN ELSE> == IDEM:< >
IDEM:<plur dat>
|   IDEM:< > == Null
Null:<plur dat>
```


https://doi.org/10.1017/S0022226798007294 Published online by Cambridge University Press

```
|   Null:<> ==
Suffix_er:<phn syl1 peak plur dat>
|   Suffix_er:<> == Suffix_e1
Suffix_e1:<phn syl1 peak plur dat>
|   Suffix_e1:<phn syl1 peak> == @
Suffix_er:<phn syl1 coda plur dat>
|   Suffix_er:<> == Suffix_e1
Suffix_e1:<phn syl1 coda plur dat>
|   Suffix_e1:<> == Suffix_n
Suffix_n:<phn syl1 coda plur dat>
|   Suffix_n:<> == Affix
Affix:<phn syl1 coda plur dat>
|   Affix:<> == Syllable
Syllable:<phn syl1 coda plur dat>
|   Syllable:<phn $yll coda> == ''<phn $yll body>''
|                                ''<phn $yll tail>''
Suffix_er:<phn syl1 body plur dat>
|   Suffix_er:<phn syl1 body> == r
Suffix_er:<phn syl1 tail plur dat>
|   Suffix_er:<> == Suffix_e1
Suffix_e1:<phn syl1 tail plur dat>
|   Suffix_e1:<> == Suffix_n
Suffix_n:<phn syl1 tail plur dat>
|   Suffix_n:<phn syl1 tail plur dat> ==
|                    IF:<AND:<POLYSYLLABLE:<>
|                              EQ:<n ''Root:<phn syl1 tail-
                                                      >''>>
|                    THEN Null
|                    ELSE n>
POLYSYLLABLE:<>
|   POLYSYLLABLE:<> == VOWEL:<Root:<phn syl2 peak->>
Mann:<phn syl2 peak->
|   Mann:<> == Noun_C
Noun_C:<phn syl2 peak->
|   Noun_C: == Noun
Noun:<phn syl2 peak->
|   Noun:<> == Word
Word:<phn syl2 peak->
|   Word:<> == Syllable
Syllable:<phn syl2 peak->
|   Syllable:<> == Null
Null:<phn syl2 peak->
|   Null:<> ==
VOWEL:<>
|   VOWEL:<> == F
Mann:<phn syl1 tail->
|   Mann:<phn syl1 tail-> == n
EQ:<n n>
|   EQ:<$x $x> == T
AND:<F T>
|   AND:<F> == F
Null:<phn syl1 tail plur dat>
|   Null:<> ==
IF:<F THEN ELSE n>
|   IF:<F THEN ELSE> == IDEM:<>
IDEM:<n>
|   IDEM:<$char> == $char <>
IDEM:<>
```


https://doi.org/10.1017/S0022226798007294 Published online by Cambridge University Press

```
|   IDEM:<> == Null
Null:<>
|   Null:<> ==
VOWEL:<@ r n plur dat>
|   VOWEL:<$vowel> == T
Mann:<phn syll tail- plur dat>
|   Mann:<phn syll tail-> == n
Mann:<phn syll tail- plur dat>
|   Mann:<phn syll tail-> == n
Devoice:<n plur dat>
|   Devoice:<> == IDEM
IDEM:<n plur dat>
|   IDEM:<$char> == $char <>
IDEM:<plur dat>
|   IDEM:<> == Null
Null:<plur dat>
|   Null:<> ==
IF:<T THEN n ELSE n plur dat>
|   IF:<T THEN> == IDEM:<>
IDEM:<n ELSE n plur dat>
|   IDEM:<$char> == $char <>                                    [4 = n]
IDEM:<ELSE n plur dat>
|   IDEM:<> == Null
Null:<ELSE n plur dat>
|   Null:<> ==
Mann:<mor suffix plur dat>
|   Mann:<> == Noun_C
Noun_C:<mor suffix plur dat>
|   Noun_C:<mor suffix plur> ==
|                          ''Suffix er:<phn root form plur>''
Suffix_er:<phn root form plur dat>
|   Suffix_er:<> == Suffix_e1
Suffix_e1:<phn root form plur dat>
|   Suffix_e1:<> == Suffix_n
Suffix_n:<phn root form plur dat>
|   Suffix_n:<> == Affix
Affix:<phn root form plur dat>
|   Affix:<> == Syllable
Syllable:<phn root form plur dat>
|   Syllable:<phn root> == <phn syl1>
Syllable:<phn syl1 form plur dat>
|   Syllable:<phn $yll form> == ''<phn $yll onset>''
|                             ''<phn $yll rhyme>''
Suffix_er:<phn syl1 onset plur dat>
|   Suffix_er:<> == Suffix_e1
Suffix_e1:<phn syl1 onset plur dat>
|   Suffix_e1:<> == Suffix_n
Suffix_n:<phn syl1 onset plur dat>
|   Suffix_n:<> == Affix
Affix:<phn syl1 onset plur dat>
|   Affix:<> == Syllable
Syllable:<phn syl1 onset plur dat>
|   Syllable:<> == Null
Null:<phn syl1 onset plur dat>
|   Null:<> ==
Suffix_er:<phn syl1 rhyme plur dat>
|   Suffix_er:<> == Suffix_e1
Suffix_e1:<phn syl1 rhyme plur dat>
```


https://doi.org/10.1017/S0022226798007294 Published online by Cambridge University Press

```
|   Suffix_e1:<> == Suffix_n
Suffix_n:<phn syl1 rhyme plur dat>
|   Suffix_n:<> == Affix
Affix:<phn syl2 rhyme plur dat>
|   Affix:<> == Syllable
Syllable:<phn syl1 rhyme plur dat>
|   Syllable:<phn $yll rhyme> == Stress ''<phn $yll peak>''
|                                    ''<phn $yll coda>''
Stress:<phn syl1 rhyme plur dat>
|   Stress:<phn $yll rhyme> == IF:<EQ:<$yll ''<phn root
                                                    focus>''>
|                              THEN ^
|                              ELSE Null>
Suffix_er:<phn root focus plur dat>
|   Suffix_er:<> == Suffix_e1
Suffix_e1:<phn root focus plur dat>
|   Suffix_e1:<> == Suffix_n
Suffix_n:<phn root focus plur dat>
|   Suffix_n:<> == Affix
Affix:<phn root focus plur dat>
|   Affix:<> == Syllable
Syllable:<phn root focus plur dat>
|   Syllable:<phn root> == <phn syl1>
Syllable:<phn syl1 focus plur dat>
|   Syllable:<> == Null
Null:<phn syl1 focus plur dat>
|   Null:<> ==
EQ:<syl1 plur dat>
|   EQ:<> == F
Null:<phn syl1 rhyme plur dat>
|   Null:<> ==
IF:<F THEN ^ ELSE plur dat>
|   IF:<F THEN $char> == <F THEN>
IF:<F THEN ELSE plur dat>
|   IF:<F THEN ELSE> == IDEM:<>
IDEM:<plur dat>
|   IDEM:<> == Null
Null:<plur dat>
|   Null:<> ==
Suffix_er:<phn syl1 peak plur dat>
|   Suffix_er:<> == Suffix_e1
Suffix_e1:<phn syl1 peak plur dat>
|   Suffix_e1:<phn syl1 peak> == @                          [5 = @]
Suffix_er:<phn syl1 coda plur dat>
|   Suffix_er:<> == Suffix_e1
Suffix_e1:<phn syl1 coda plur dat>
|   Suffix_e1:<> == Suffix_n
Suffix_n:<phn syl1 coda plur dat>
|   Suffix_n:<>Affix
Affix:<phn syl1 coda plur dat>
|   Affix:<> == Syllable
Syllable:<phn syl1 coda plur dat>
|   Syllable:<phn $yll coda> == ''<phn $yll body>''
|                                ''<phn $yll tail>''
Suffix_er:<phn syl1 body plur dat>
|   Suffix_er:<phn syl1 body> == r                          [6 = r]
Suffix_er:<phn syl1 tail plur dat>
|   Suffix_er:<> == Suffix_e1
```


https://doi.org/10.1017/S0022226798007294 Published online by Cambridge University Press

```
Suffix_e1:<phn syl1 tail plur dat>
|   Suffix_e1:<> == Suffix_n
Suffix_n:<phn syl1 tail plur dat>
|   Suffix_n:<phn syl1 tail plur dat> ==
|                 IF:<AND:<POLYSYLLABLE:<>
|                           EQ:<n ''Root:<phn syl1 tail->''>>
|                 THEN Null
|                 ELSE n>
POLYSYLLABLE:<>
|   POLYSYLLABLE:<> == VOWEL:<Root:<phn syl2 peak->>
Mann:<phn syl2 peak->
|   Mann:<> == Noun_C
Noun_C:<phn syl2 peak->
|   Noun_C:<> == Noun
Noun:<phn syl2 peak->
|   Noun:<> == Word
Word:<phn syl2 peak->
|   Word:<> == Syllable
Syllable:<phn syl2 peak->
|   Syllable:<> == Null
Null:<phn syl2 peak->
|   Null:<> ==
VOWEL:<>
|   VOWEL:<> == F
Mann:<phn syl1 tail->
|   Mann:<phn syl1 tail-> == n
EQ:<n n>
|   EQ:<$x $x> == T
AND:<F T>
|   AND:<F> == F
Null:<phn syl1 tail plur dat>
|   Null:<> ==
IF:<F THEN ELSE n>
|   IF:<F THEN ELSE> == IDEM:<>
IDEM:<n>
|   IDEM:<$char> == $char<>                                    [7 = n]
IDEM:<>
|   IDEM:<> == Null
Null:<>
|   Null:<> ==
Mann:<mor word plur dat> = m ^ E n @ r n
```


https://doi.org/10.1017/S0022226798007294 Published online by Cambridge University Press