CELEX: Building a Multifunctional, Polytheoretical Lexical Database

Ton van der Wouden

0. Abstract

Recent developments in Computational Linguistics have brought about an increasing interest in large scale lexical modules, at a time when current trends in hardware and software engineering bring this goal within reach. This paper describes one such system, the CELEX database. For expository purposes only, this system is contrasted with another big project that starts from different premisses and is meant to serve different goals[1].

1. Aims

Researchers in the fields of, amongst others, theoretical linguistics, psycholinguistics, machine translation and natural language interfaces, are clearly in need of tools to supply them with both on-line facilities and off-line applications with regard to collecting information on lexical data. At this moment, the only available sources of lexical material are dictionaries (printed or machine-readable), text corpora, word lists and frequency lists. These, however, tend to be inflexible and using them is often time-consuming and error-prone. Furthermore, systematic searches for (sets of) words that are related as to one or more properties are close to being impossible.

In the world of computing developments are fast. The money you spent last year to obtain the finest, fastest new machine would be worth much more now: more speed, more memory, more advanced technology, more user-friendliness for the same amount of money. Developments in the software field are perhaps not as dramatic as those in hardware, but they are still impressive: new compilers, new programming languages and even new concepts in programming, new software development tools and special-purpose programs come on to the market every day. For manufacturers the struggle for life is tough: economic laws force them to decrease their prices and to spend most of their profits on research and development.

A few years ago the Dutch government decided that developments in hardware and software made it realistic to try and fulfill the needs of the research community by funding the development of a Centre for Lexical Information (CELEX). CELEX is a conjoint initiative of five research institutes in the Netherlands, viz. the University of Nijmegen, the Institute for Dutch Lexicology (INL) in Leyden, the Max-Planck-Institute for Psycholinguistics in Nijmegen, the Institute for Perception Research (IPO) in Eindhoven and the Dr. Neher Research Laboratory of the Dutch Telephone Company (PTT) in Leidschendam. CELEX is carried out within the Interfaculty Research Unit for Language and Speech (IWTS) of Nijmegen University. The aim of the project is to make available computerized, **multilingual,**

multifunctional, and **polytheoretical** lexical databases to interested institutions and companies via modern electronic access methods based on the Dutch research network (SURFNET).

The CELEX database will be operational from January 1989 onwards[2]. It will then contain the following information, both for Dutch and English:

— orthographic information: graphemes, hyphenation positions, variants, accents etc.

— phonological information: phonemes, allophones, syllable structure, stress, uniqueness point[3] etc.

— morphological information: hierarchical decomposition into free and bound morphemes, inflectional paradigms, morphemic relations etc.

— syntactic information: grammatical word class, grammatical valency, inflectional attributes etc.

— frequency information: per word form, lemma, morpheme etc., based on recent and representative text corpora[4].

For obvious reasons, CELEX will not deal with semantic information during the first developmental stage: there is no semantic theory that is unambiguous, precise and general enough to describe, in an interesting and implementable way, the semantics of so many words and all the relations between them.

A first prototype of the Dutch database was made accessible in December 1987, whereas the first, preliminary, English version will be released in September 1988. In what follows, we will discuss some major aspects of CELEX, and we will contrast the CELEX approach with lexicon system developed by Marc Domenig that is destined to be part of the EUROTRA machine translation system of the European Community.

1.1 Multilingual

The database is multilingual in nature. When the CELEX system is in its final state, the entire system will consist of a number of monolingual databases (for the time being Dutch and English), which are structured as parallel as possible. At the beginning of the project, in 1986, most work was devoted to the contents of the Dutch database. Simultaneously, attention was focused on the actual design of the database system that was to hold all the lexical data as efficiently as possible. From a theoretical point of view, the relational model is the most interesting. Moreover, "In fact, there can be little doubt that the relational approach represents the dominant trend in the marketplace today, and that 'the relational model' [. . .] is the single most important development in the entire history of the database field."[5] Research made it evident that the ORACLE Relational Database Management System would suit our purposes best[6]. The system was implemented and refined so as to be optimally adapted to our specific wishes.

The Dutch part of the CELEX database is almost completed now, resulting in a version containing detailed information on orthography, phonology, morphology, syntax and word frequencies for more than 100,000 stems and over 300,000 inflected forms. Detailed work on the English database has been well underway for several months now, with a view to constructing a database that is, as far as

possible, identical to the Dutch counterpart of the project, ensuring that while the English database is internally consistent and independent, the end result will be as highly flexible and sophisticated as the Dutch database. Plans are also being made for extensions to German and French.

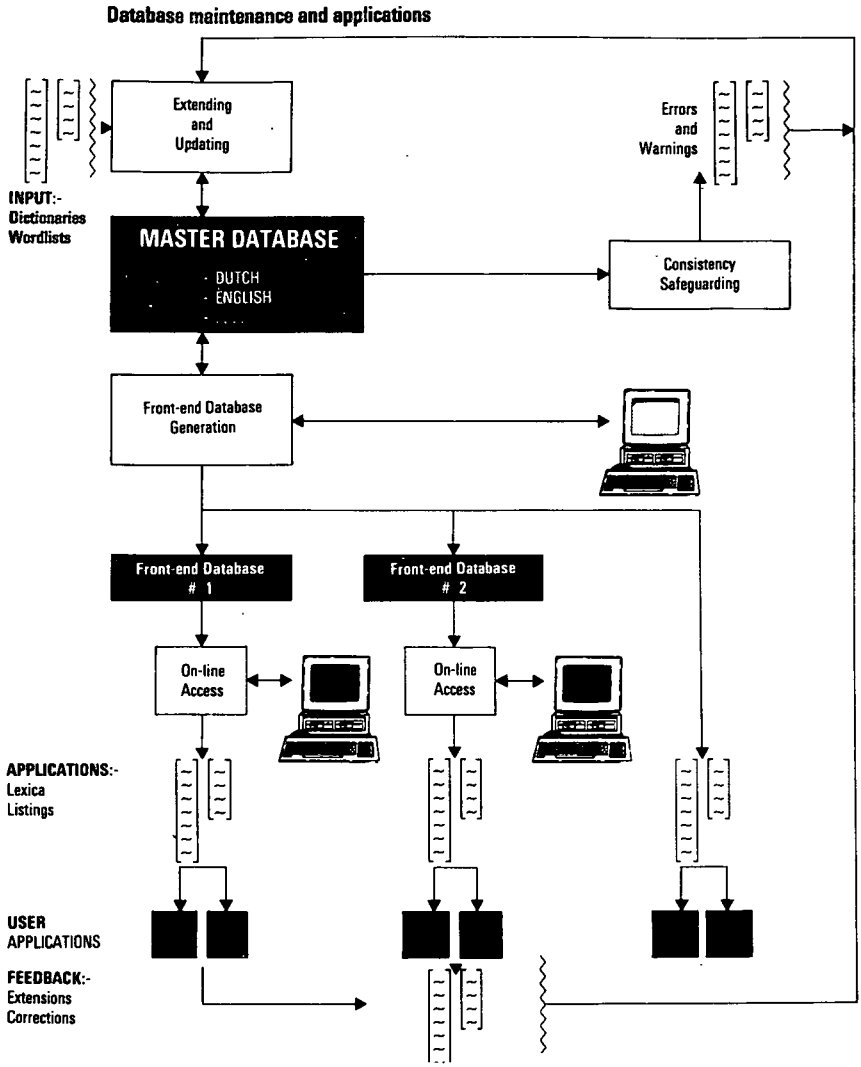
1.2 Multifunctional

Not only is the database multilingual, it is multifunctional in nature as well. Among the people and institutions interested in our project are researchers from various disciplines (phoneticians, psycholinguists, theoretical linguists, machine translation experts, etc.). In order to guarantee optimal flexibility, the database has been designed in such a way that the entire set-up of the master database, together with a user interface, allows the user to define his/her own questions and subsequently retrieve a virtually infinite number of so-called front-end databases. The user is able to specify his/her database—or application lexicon—by selecting the proper data, formats and conditions from a hierarchically organized set of menu options with various on-line help facilities. Along with the database itself, a powerful full-screen application is generated automatically, which can be used to query the resulting database. Clearly, a main advantage of the user interface facilities is that, in principle, any query can be performed without having to master the underlying SQL database management language — although this is available. Finally, these facilities will include options to generate listings and files in different formats, so that the selected lexical data can be employed in applications outside the ORACLE DBMS, and on other computers[7].

1.3 Polytheoretical

Finally, the database is also polytheoretical (this adjective was, as far as we know, coined during the 1987 Stanford Linguistic Institute Lexicon Workshop), by which we mean that the information in it is both rich and flexible enough to allow researchers working within the various current theoretical frameworks to extract the information they need in the format they prefer. For example, for a parser based on the Government and Binding framework[8] another type of parsing lexicon, with different entries, will be needed than for a parser based on some variant of Categorial Grammar[9]. CELEX will be able to offer both, and others as well.

1.4 Overview of the system*



*From the CELEX Newsletter.

2. Regularities in the Lexicon

In older linguistic theories[10] the lexicon was seen as a receptacle where all irregularities of the language were stored, whereas the grammar expressed the regular part of language. In current linguistic theories, however, the lexicon is no longer seen as an appendix to grammar. On the contrary: a major trend in linguistics since the early 70's has been the recognition that the lexicon should be viewed as the major store of linguistic information. Linguistic and other rules govern both the information stored with lexical elements and the relations between elements. For instance, there are relationships between the spellings of words and their pronunciations; between morphological structure and stress pattern; between morphological structure and meaning.

This kind of regularity, postulated in the mental lexicon, is reflected in computer lexicon systems. Lexicons for computer applications are no longer simple lists of words, stored in a more or less efficient way, but complex, hierarchical systems[11] with internal structure. In what follows, we will describe two such systems: a special purpose lexicon system and the general purpose CELEX approach.

3. Approaches to Lexicon Systems

In a series of recent publications[12] Marc Domenig develops a lexicon component for Eurotra, the machine translation project of the Eurpean Community. More precisely it is a proposal for special purpose software for lexical matters associated with this MT-project. The following three demands are adopted as design criteria for his dictionary formalism[13]:

1. Linguistic felicity: the formalism should be as close as possible to known linguistic notations.
2. Expressiveness: the formalism should be adequate in power, i.e. powerful enough to cover the targeted poblems.
3. Computational efectiveness: there should be efficient computational devices to interpret the facts expressed by a 'program' written with the formalism.

There is no reason not to agree with these demands, but they are of so general a nature that a range of possible implementations may be thought of, as we will show below.

3.1 Domenig

In Domenig's system,

> "a dictionary is redefined to comprise a 'dynamic' component, which both extends and partly replaces the information stored in the purely 'static' entries of a traditional dictionary. The extension of the information content is achieved by integrating knowledge about linguistic processes which can be executed on a computer. Intelligently conceived, such processes will eliminate much of the redundancy encountered in traditional dictionaries, thus improving not only the information content but also the conceptual structuring."[14]

The processes modelled in his lexicon system are presently focussing around morphology[15]. The first step taken is to try and provide the means to define the structures and regularities of the graphological identifiers for words, i.e. their string transcriptions. To express this information, Domenig's system is built around a morphological module structured along the so-called Two-level model[16]. This model, which is rather popular nowadays in the computational linguistics world, is not a kind of new morphological theory, but a formalism that shows, from a computational point of view, quite some resemblance to an ALGOL-68-like programming language.

The following advantages of the model have been claimed[17].

1. Independence of the object language: the morphological processes of various languages have been implemented in the system.
2. Power and problem orientation: because of the parallel applications of the rules, the system is relatively easy implementable and extendable; the rules are declarative.
3. Linguistic felicity: the rules are easily understood by linguists, because very similar ones have been used for years in structural phonology (like in SPE[18]).
4. Bidirectionality: the rules can both be used for generation and for analysis of forms.
5. Efficiency: "The computational effectiveness of the two-level model compares very favourably with other natural language processing devices."[19]

3.2 CELEX

In the CELEX database, the use of rules and regularities is quite different. Compared to the dynamic system of Domenig, CELEX could be called static. All information in the database is explicitly stored. In order to derive and guard this information, however, extensive use of rule-based systems is made. We give an example in the next section.

In order to derive morphological analyses for all the Dutch words in the database, a morphological analyzer is developed[20]. Hundreds of thousands of words were analyzed by the program; about 80% received one or more tentative analyses. In a post-editing phase the output of the program was extensively scanned: all words were checked by hand: analyses were corrected, added and/or deleted, resulting in a file of complete morphological analyses for (almost) all Dutch words.

Together with all kinds of other information, this file of morphological analyses has been implemented in the relational database management system. There again linguistic and other regularities came into play. An example: usually, the phonological representation of a Dutch compound is easily derivable from the phonological representation of its parts, as is shown below:

WORD	PHONOLOGY	MORPHOLOGY	SEMANTIC
appel	Ap&l		'apple'
azijn	azEIn		'vinegar'
appelazijn	Ap&l#azEIn	((appel), (azijn))	'apple vinegar'

In order to check the information stored the output of rules was computed and the computed forms were compared with the stored forms. In a way, this checking process is comparable with the function of Feature Specification Defaults (FSDs) known from the GPSG[21] framework.

A system like this shows a lot of redundancy; the representation may hardly be called efficient. In due time, things might change: if the information stored in the database is found reliable enough, the part that is derivable by rule could be stored dynamically. Again, this is comparable with a mechanism known from GPSG: this dynamic storage of regular information can be seen as a kind of usage of the Feature Specification Default (FSD) mechanism.

There could, however, be quite a problem with dynamic storage of lexical data in the CELEX database: to answer whole classes of relevant questions this way of storing information is extremely inefficient. To illustrate our point, we again have to take a closer look at morphological information.

According to recent morphological theory, the morphological module of the language system shows various levels; in other words, according to modern insights, (groups of) morphological rules are ordered[22]. For Dutch (as for English) two groups of suffixes are postulated. The first group is supposed to be attached before application of stress rules, the second afterwards. As the examples show, this amounts to stress shifting as opposed to stress neutral affixes: stress shifting suffixes bear primary word stress[23]:

Suf1 (stress shifting)		
mil'jard + air	miljar'dair	"billionaire"
Suf2 (stress neutral)		
'rood + achtig	'roodachtig	"reddish"

The difference between the Level I and Level II affixes in Dutch can be motivated on other grounds (syllabification, readjustment) as well, but that is irrelevant here. Much more interesting is, that the ordering hypothesis for morphological rules predicts that certain structures are possible while others should be ruled out:

[[[x]Suf1]Suf2] *[[[x]Suf2]Suf1]

At first sight, this hypothesis does the right predictions: **miljardairachtig,** showing the lefthand structure, is a well-formed word, whereas ***roodachtigair,** of the righthand structure, is terrible. Extensive testing this hypothesis however, that is, seriously trying to find counterexamples (of the righthand form), might be very cumbersome in a dynamic lexicon system. There are two possibilities: either the Ordering Hypothesis is part of the rules that define the morphological information in the system, or it is not. In the first case, no counterexample will be found because the system cannot generate the analysis we are looking for since the Ordering Hypothesis rules it out. Not finding a counterexample can, in this case, hardly be seen as corroboration of the level ordering hypothesis: it is just a demonstration of circularity. In the second, a counterexample might or might not be found, but at what a cost: for every word in the system, the morphological analysis or analyses should be computed in order to see whether it is a counterexample or not[24].

In a static system, however, questions like the one discussed above are relatively easy to handle: the analyses of all words in the system are derived once, checked and debugged extensively, and stored efficiently. Retrieval is fast and easy.

4. Discussion

Given the current state of the art in computer lexicon research, there seems to be a tension between efficiency of storage and lack of redundancy on the one hand, and reliability, i.e. quality of the information, on the other. Rules can be used and are needed even to efficiently generate and implement lexicon information, but the quality of such a lexicon is at most as high as the quality of the rules used. The only way to test the rules, and to lay hands on the necessary lists of all exceptions to the rules, seems to be to build a classical, static lexicon system, where all information is stored at length.

Another way of approaching and explaining the differences in the systems described is from the perspective of the purpose of the lexicon system and the things it should be able to do. The Domenig system is an important but small part of a huge machine translation system; the sort of questions that the system should be able to answer are rather clear from the start, the theoretical framework and the formalism to express questions and answers are fixed. On the other hand, the CELEX system is, as we have seen, meant to be multifunctional. It is supposed to function both as a very general research tool to help researchers answer as many lexical questions as possible, and as a sort of mother lexicon from which special purpose lexicons can be derived. The questions the researcher will ask are unpredictable, just as the theoretical framework he/she is working in and the formalism he/she will prefer to express his/her questions in and he expects his/her answers.

Again another way to describe the differences between the two approaches can be related to the following quotation from Domenig[25].

> "The internal implementation and organisation of this [lexical] material on a computer, however, must by no means be trivial; in contrast to the designers of traditional dictionaries, who are bound to the sequential nature of printed media, we are able to abstract the surface representation in a computationally implemented dictionary: taking advantage of the computer's processing abilities, we may structure the information internally as we like, provided that we can define a suitable mapping function to the surface representation. If the internal structure is well conceived, we may even define **different** mapping functions, thus realising e.g. a conventional dictionary, a thesaurus etc. with the same data base."

Both Domenig's system and our's follow this strategy[26], but Domenig's lexicon tool is designed more in the direction of (EUROTRA) translation; we predict that, all other things being equal, it will be simpler to derive a lexicon module for MT purposes from Domenig's system than from the CELEX database. However, it will be much harder to derive special purpose lexicons for other applications from his system than from our's, and the more so if this other application has demands that are more different from the specifications used in a MT-dictionary.

In other words: the way one implements one's lexicon depends on its applications, i.e. on the range of questions one wants to be able to answer. With bulk memory devices becoming cheaper almost by the day, efficiency of storage becomes a less important argument in the discussion. The stress shifts to efficiency of retrieval, and looking things up in an efficient architecture seems, in general, to be faster than computing things.

A reasonable question to end our paper with is whether one can completely do without rules, that is, without a dynamic part of the system. The answer should probably be negative: in languages such as Dutch and German one can, in principle, make an infinite number of compounds, in languages such as Finnish the number of inflected forms is extremely high, too high to fully store all forms.

Note however that recent psycholinguistic research (Jarvella *et al.* 1987, as cited by Schreuder 1987) suggests that the mental lexicon of speakers of morphologically complex languages is organized in a different way than the mental lexicon of speakers of morphologically simple languages: in speakers of Dutch, inflected forms seem to be stored as such, whereas they are derived in speakers of Italian. This suggests that if the regularities of the mental lexicon are to be reflected in the lexicon system, a dynamic structure should be chosen for a lexical database for Italian, whereas a static structure is more applicable for its Dutch counterpart. In other words, even at the level of database architecture, language independency is not something to aim for. One might therefore want to infer that it is impossible to build lexical databases for these two (Indo-European) languages in parallel. And this could lead to the more general conclusion that it is, in principle, impossible to build a multilingual lexical database as we have defined it if the languages to be covered are too different.

Notes

[1] Thanks are due to Dirk Heylen en Domien Kusters for proofreading and discussions. All errors are of course my own.

[2] See CELEX Newsletter for more information.

[3] According to Schreuder & Kerkman (1987) the uniqueness point is "that point in time at which a word can be recognized from the acoustic information, [...] that point [...] at which its initial sequence of phonemes is common to that word and no other".

[4] For Dutch, the Leyden INL-corpus (> 45,000,000 tokens) will be used, for English the Birmingham COBUILD corpus (± 20,000,000 tokens).

[5] Date (1986: 20).

[6] Van der Veer, Wittenburg & Kerkman (1986). ORACLE is a trademark of Oracle Corporation, CAL, USA.

[7] Extensions of the query language SQL and the user interface of the ORACLE database management system with more powerful string operation functions are under development.

[8] Chomsky (1981).

[9] Moortgat (1988), chapter 7 of Van Benthem (1986).

[10] As in Bloomfield (1933).

[11] The standard reference for efficient storage and retrieval of data is Knuth (1973). On computer lexica: Domenig (1986, 1987), Domenig & Shann (1986), Calder (1988), Walker *et al.* (eds.) forthcoming.

[12] Domenig (1986, 1987), Domenig & Shann (1986).

[13] After Domenig (1986).

[14] Domenig (1986).

[15] The same holds for Calder's PROTOLEXICON system, cf. Calder (1988).

[16] Koskenniemi (1983), Karttunen (1983), Dalrymple e.a. (1987).

[17] Domenig (1987).

[18] Chomsky & Halle (1968).

[19] Domenig 1986. Cf. however note 24 below.
[20] Van der Wouden (1988).
[21] Gazdar, Klein, Pullum & Sag (1985).
[22] Scalise (1984).
[23] Scalise (1984: 89).
[24] Note that, although Two-level-rules may be compiled into a Finite State Machine (Karttunen 1983, Dalrymple *et al.* 1987), the Two-level-system is of very high computational complexity (Domenig 1986, Barton *et al.* 1987).
[25] Domenig (1986).
[26] As do many others.

References

Barton, E., R. Berwick, and E. Ristad 1987. *Computational Complexity and Natural Language*. Cambridge, MIT Press.

Benthem, J. van. 1986. *Essays in Logical Semantics*. Dordrecht [etc.]: Reidel.

Bloomfield, L. *Language*. 1933. New York: Holt.

CELEX Newsletter. Nijmegen, 1987-.

Calder, J. 1988. "Polytheoretic Lexicons and Reusable Dictionaries". MS. Edinburgh.

Chomsky, N. and M. Halle. 1968. *The Sound Pattern of English*. New York: Harper & Row.

Chomsky, N. 1981. *Lectures on Government and Binding*. Dordrecht [etc.]: Foris.

Dalrymple, M., R. Kaplan, L. Karttunen *et al.* 1987. 'DKIMMO/TWOL: A Development Environment Morphological Analysis'. Ms. Xerox PARC/CSLI.

Date, C. 1986. *An Introduction to Database Systems*. Vol. 1, 4th. ed. Reading, Mass., [etc.]: Addison-Wesley.

Domenig, M. 1987. 'On the Formalisation of Dictionaries'. *in Sprache und Datenverarbeitung 1*, 36—41.

Domenig, M. 1987. *Entwurf eines dedizierten Datenbanksystems für Lexika. Problemanalyse und Software-Entwurf anhand eines Projektes für maschinelle Sprachübersetzung*. Tübingen: Niemeyer.

Domenig, M. and P. Shann. 1986. 'Towards a Dedicated Management System for Dictionaries' in *Proceedings of Coling '86*. 91—96.

Gazdar, G., E. Klein, G. Pullum and I. Sag. 1985. *Generalized Phrase Structure Grammar*. Oxford: Basil Blackwell.

Jarvella, R. J., R. Job, G. Sandstrom *et al.* 1987. 'Morphological Constraints on Word Recognition' in A. Allport, D. Mackay, W. Prinz. *et al.* (eds.). *Language Production and Perception*. London: Academic Press.

Karttunen, L. 1983. "KIMMO: A Two-Level Morphological Analyzer" in *Texas Linguistic Forum* 22. 253—270.

Koskenniemi, K. 1983. *Two-Level Morphology: A General Computational Model for Word-Form Recognition and Production*. Helsinki: University of Helsinki (Publications University of Helsinki, Department of General Linguistics, 11).

Knuth, D.: *The Art of Computer Programming*. Vol. 3. Sorting and Searching. Reading, Addison-Wesley, 1973.

Moortgat, M. 1988. *Categorial Investigations. Logical and Linguistic Aspects of the Lambek Calculus*. Dordrecht [etc.]: Foris.

Scalise, Sergio. 1984. *Generative Morphology*. Dordrecht [etc.]: Foris.

Schreuder, R. 1987. *Het mentale lexicon*. Nijmegen: Katholieke Universiteit.

Schreuder, R. and H. Kerkman. 1987. 'On the Use of a Lexical Database in Psycholinguistic Research' in W. Meijs (ed.). *Corpus Linguistics and Beyond*. Amsterdam: Rodopi.

Veer, C. van de, P. Wittenburg, and H. Kerkman. 1986. *Logical Structure of a Lexical Database and Selection of an Appropriate DBMS*. Nijmegen: Max-Planck-Institut für Psycholinguistik.

Walker, D., A. Zampolli & N. Calzolari (eds.): *Automating the Lexicon: Research and Practice in a Multilingual Environment*. Cambridge, Mass.: The MIT Press, forthcoming.

Wouden, T. van der. 1988. 'Automatic Morphology for Lexical Databases' in *Gramma* 12: 1.