

1996

BG-18

mote interne

IST. EL. INF.  
BIBLIOTECA  
Posiz. ARCHIVIO

ET-10/51  
Deliverable 8

BG-18  
1996

### Evaluation Report

**Geoff Barnbrook, Nicoletta Calzolari, Stefano Federici,  
Martin Hoelter, Simonetta Montemagni, Carol Peters,  
Helmut Schnelle, John Sinclair**

**Sprachwissenschaftliches Institut  
Ruhr-Universität Bochum  
D-44780 Bochum  
Federal Republic of Germany**

# Contents

<b>1</b>	<b>The Methodology Perspective</b>	<b>1</b>
1.1	The Birmingham approach . . . . .	1
1.2	The Pisa approach . . . . .	3
1.2.1	The first stage . . . . .	3
1.2.2	The second stage . . . . .	4
1.2.3	The third stage . . . . .	5
1.3	The Bochum approach . . . . .	6
1.3.1	General assumptions . . . . .	6
1.3.2	The mapping strategy . . . . .	7
<b>2</b>	<b>The Users' Perspective</b>	<b>10</b>
2.1	Birmingham—The parser in the practice of lexicography . . . . .	10
2.2	Pisa—Applying the results of the Pisa syntactic-semantic parser . . . . .	11
2.2.1	Intermediate results . . . . .	11
2.2.2	Final results . . . . .	12
2.3	Bochum—The implications of d21 . . . . .	13
<b>3</b>	<b>The Theory Perspective</b>	<b>15</b>
3.1	Pisa—What are the theories implied by the Pisa representations? . . . . .	15
3.1.1	Combining theory and practice . . . . .	15
3.1.2	Integrating syntactic and semantic information . . . . .	15
3.1.3	Constraints or preferences? . . . . .	16
3.1.4	Two different approaches in the construction of the type system . . . . .	16
3.2	Bochum—The Cobuild-BLF-HPSP correlation . . . . .	17
3.2.1	Evaluation criteria . . . . .	17
3.2.2	Contextual information and semantic types . . . . .	17
<b>4</b>	<b>The Alep Perspective</b>	<b>21</b>
4.1	The Bochum and Pisa representations of Cobuild lexical entries relative to the Alep formalism . . . . .	21
4.1.1	What can be represented in ALEP . . . . .	21
4.1.2	What cannot be represented in ALEP . . . . .	23
4.1.3	Inheritance in ALEP . . . . .	23
4.1.4	Summary . . . . .	24
4.2	Reusability of results in existing ALEP grammars . . . . .	24
<b>5</b>	<b>The Corpus Perspective</b>	<b>27</b>
5.1	Birmingham—Cobuild definitions and technical vocabulary . . . . .	27
5.2	Pisa—Testing the Pisa representations on the ITU corpus . . . . .	32
5.3	Bochum—Cobuild-based entries and ALEP grammars . . . . .	36

# 1 The Methodology Perspective

In this section the methods of extraction of linguistic information from the definition text of the Cobuild Student's Dictionary will be examined. There are three subsections, one dealing with the approach of each of the partners.

## 1.1 The Birmingham approach

The language of dictionary definition has been for centuries a highly specialised form, related to English but with substantial variations from normal writing.

Having the character anciently ascribed to the planets, wandering; erratic; as, a *planetary career*

To go in a gallop, as a horse ... Quadrupedal motion by a regular succession of leaps

[*Practical Standard Dictionary*. London: Funk and Wagnall, 1925]

Some recent projects have provided analyses of this kind of language to aid knowledge extraction from conventional dictionaries. The growing awareness of "reusability" as a concept in language technology gave strength to the task, and made it worthwhile for special extraction tools to be developed.

In contrast to all other dictionaries, the Cobuild definitions are written in ordinary English sentences; a tool that can parse these is a partial parser of English, a way of interpreting the structure of the sentences that establishes a systematic relationship with the other sentences of English. Reusability is further pursued in that the effort expended on the construction of the parser is a contribution to the larger task of parsing the whole language adequately.

From the outset Birmingham regarded the definition text as a set of sentences in a sublanguage. A sublanguage is an important concept in natural language processing; it is a variety of a language which has two special characteristics:

- a. It relates to a homogeneous specialised area of human activity and hence communication.
- b. It forms a coherent subset of the language as a whole.

It is assumed that by narrowing the subvariety, usually in a technical context, the actual structure of a language will simplify, and thus become more amenable to automatic processing. A sublanguage is thus defined simultaneously by internal and external criteria, but the internal criteria are crucial; if the sublanguage is not very substantially simpler than the whole language, then the narrowing of focus does not aid the parsing task.

It was clear on cursory examination of the definitions that not all the structures of English were involved. For example there were only declarative sentences, so no need to worry about interrogatives or imperatives. There was therefore no need to write or borrow a comprehensive parser for English.

Further study revealed that the restrictions on the syntax were very substantial, and a new fact emerged—the most efficient parser for the sublanguage was not necessarily a subset of the parser for the general language. So specialised were the definitions that a special grammar was written for them. At some points it overlaps with a general grammar, but particularly in the more abstract statements, the functions of the definitions take precedence over the superficial similarity to English as a whole.

It may be, of course, that this observation merely reflects the somewhat primitive state of automatic parsing at the present time. A thorough and comprehensive parser might well recognise a definition statement as different from other kinds of statement, and incorporate some or all of the sublanguage parser we have written. Certainly we would have been misguided to have made the initial assumption that a general language parser would be both adequate and adapted to our needs.

It would be inadequate because we were obliged to recognise several major structural features which are not available in current parsers, for example *matching*. In the definitions, certain words and phrases that occur in the earlier part of the definition statement are to be matched with later occurrences of predictable items. This is a criterion of well-formedness akin to pairs like *not only ... but also*, but unique to this sublanguage. In certain circumstances the matching does not take place and that circumstance affects the identification of the *definiendum* itself.

As to relevance, we find that the overall structure of most of the definition sentences can be expressed as:

*headword-in-context*, explanation

This is a structural statement that has not been offered before for ordinary English parsing. It is related to subject/predicate, to theme/rheme and to several other familiar parse categories, but it is truly none of them. If we had started with an existing parser we would have uncritically chosen a variety of clause and sentence structures that can realise this generalisation, but would not have been led to find or express it. The parser would have been much more complicated and much less adequate.

The methodology of extraction for Birmingham, then, is built into the design of the grammar that is implemented in the parser. For example, a number of headwords have restrictions on their usage which are realised by words or phrases placed immediately in front of the headword. An important subcategory of these items is a noun with a possessive affix, as in "a bird's beak". Two classes can then be prepared, namely those headwords that suffer such restrictions, and those nouns that can by occurring in that structure restrict the headword. It happens that in most cases the two words are uniquely associated with each other, so there is considerable doubt as to the value of identifying the classes, whereas in general grammar we would expect classes with much more freedom of the members to co-occur.

Structure and function coincide here, and in most other places in the grammar. This is a huge simplification compared with normal grammars. There is no need in the sublanguage grammar for two different sets of terms, one for the structures and one for the functions, because the flexibility to combine structural elements in different functions is not available.

Absolute and relative position are major determinants of structure/function. Common words like *is* and *of* do different things according to their place in the definition, even though their role, as seen by a conventional grammar, is the same. So again it would have been misleading to have begun with a conventional grammar; there are no warnings of likely variance of this kind.

In addition to the kinds of functions that are reasonably expected in grammar writing; the study of the definitions brought out several strands of meaning that are usually thought to be inferential rather than directly structural. For example when the headword is a verb, the definition structure may express a subject for that verb. The subject may belong to one of several classes, of which a small subset are interesting from a quasi-inferential point of view. These are *you* and *someone*. The choice of *you* indicates that the action of the verb is something considered normal in social behaviour; one would not be ashamed or embarrassed to engage in this activity. *Swim, think, forgive* are examples. But the choice of *someone* as subject indicates that the average person would prefer to create a distance between himself or herself and the activity; as if to say that such things are done but people like us do not do them. Verbs such as *burp, cheat, gloat* refer to actions that are prejudged as undesirable, and have *someone* as their subject.

The wording of the definitions is so carefully organised that distinctions like the above can be associated directly with structural choices. Perhaps more general grammars will one day find that a choice of this nature can be built in as part of a larger picture, and show that there is more overlap between the sublanguage grammar and the general grammar than can be claimed at the present time. Certainly, as the work of the project partners makes clear, for many of the central sources of meaning in the sublanguage there were no central categories in existing formal grammars for them. They are inserted often as fairly marginal categories, set apart from the routine choices like *tense* and *number*, which in turn are unimportant in the sublanguage.

The basis of an evaluation of Birmingham's work from the perspective of methodology will be the form of the grammar itself; its size and simplicity (given that it is deemed to be adequate for its job), the novel distinctions in meaning it makes, the different emphases of this grammar from general grammars.

## 1.2 The Pisa approach

The working strategy adopted by Pisa involved a three stage approach as follows:

- Analysis of the parsed Cobuild definitions provided by Birmingham and extraction of syntactic-semantic, collocational and lexical information;
- Evaluation of the different types of information extracted with respect to its representability and its utility for NLP;
- Conversion in a TFS formalism.

### 1.2.1 The first stage

In the first stage, samples of the definition statements were studied in depth in order to identify the different types of information contained, and the ways in which it had been represented in the dictionary. The Cobuild definition is unique in that information is encoded consistently not only on the meaning of the headword (in the right hand side—RHS) but also on its usage (the left hand side—LHS). It was decided to concentrate our efforts mainly on the extraction of information from the LHS as it was felt that much of the information contained in this part of the definition on the syntactic and semantic/lexical constraints and preferences of the arguments of a lexical item would be very useful for NLP applications. From our first analysis of the data, it became clear that the very regular structure and defining formulae employed by Cobuild to encode this kind of implicit data could be exploited in order to extract it.

A specialised parser was designed and developed to extract the information and to map it onto an Intermediate Template (see Deliverables 4 and 6 for description and examples of templates for different grammatical categories). The decision to employ this Intermediate Template (IT) was motivated by the need to be able to store (in an explicit, interpreted way) all the information extracted in a computationally tractable fashion, so that information that is not immediately representable in TFS or in ALEP can be kept for future analysis and exploitation, if and when desired. Below we give an example of the results of our analysis of apply 4, at this stage.

The definition for this sense of apply in the dictionary is: "If you apply a rule, system, or skill, you use it in a situation or activity". Our parser currently gives the following output on the IT:

```

def_no          : 1103
sense_no        : 4
def_type        : 1
lemma           : apply
entry_info      : entry           : apply
                  norm_entry      : apply
genus_info      : prov_superordinate1 : use
                  isa1             : use
                  genus_prep1     : in
inflection      : apply applies applying applied
gram            : VB with OBJ
voice           : active
inference       : possible likely
subj_info       : subj_features1   : human
obj_info        : obj3             : specific: skill
                  obj_features3   : inanimate,+count
                  obj2            : specific: system
                  obj_features2   : inanimate,+count
                  obj1            : specific: rule

```

---

	obj_features1	: inanimate,+count
usage_info	: formality	: normal
	style	: normal

### 1.2.2 The second stage

The second stage was a careful evaluation of the information extracted and stored on the Intermediate Template before implementing the TFS representation. The information was classified on the basis of whether:

- i. it is immediately representable in TFS;
- ii. it needs further analysis;
- iii. it is of scarce relevance from the linguistic viewpoint.

As far as i. is concerned, not all the information extracted from Cobuild entries can or should be represented in TFS terms. This is due either to limitations of the formalism, or— more generally—to the usefulness and exploitability of this information by NLP systems, and particularly by HPSG-like grammars. The second case refers to the so-called 'intermediate results', that is when a final and reliable semantic interpretation has not yet been assigned; in these cases, further analysis is needed before the information can be converted into a TFS form. Finally, case iii. refers to information that has been used to drive the extraction procedure or that represents an access key to the TFS lexicon we are building starting from the Cobuild dictionary.

In the following example, we can see how much of the information that had been extracted from the definition statement for **apply 4** and mapped onto the IT has been converted into the TFS formalism.

PHON	<i>&lt;apply&gt;</i>															
	CAT	<table border="1"> <tr> <td>HEAD</td> <td> <table border="1"> <tr><td>MAJOR</td><td><i>verb</i></td></tr> <tr><td>VFORM</td><td><i>bse</i></td></tr> <tr><td>PREF-VFORM</td><td><i>active</i></td></tr> </table> </td> </tr> <tr> <td>SUBJ</td> <td><i>&lt; NP : 1 [ human ] &gt;</i></td> </tr> <tr> <td>COMPS</td> <td><i>&lt; NP : 2 [ skill V system V rule ] &gt;</i></td> </tr> <tr> <td>LEX</td> <td>+</td> </tr> </table>	HEAD	<table border="1"> <tr><td>MAJOR</td><td><i>verb</i></td></tr> <tr><td>VFORM</td><td><i>bse</i></td></tr> <tr><td>PREF-VFORM</td><td><i>active</i></td></tr> </table>	MAJOR	<i>verb</i>	VFORM	<i>bse</i>	PREF-VFORM	<i>active</i>	SUBJ	<i>&lt; NP : 1 [ human ] &gt;</i>	COMPS	<i>&lt; NP : 2 [ skill V system V rule ] &gt;</i>	LEX	+
HEAD	<table border="1"> <tr><td>MAJOR</td><td><i>verb</i></td></tr> <tr><td>VFORM</td><td><i>bse</i></td></tr> <tr><td>PREF-VFORM</td><td><i>active</i></td></tr> </table>	MAJOR	<i>verb</i>	VFORM	<i>bse</i>	PREF-VFORM	<i>active</i>									
MAJOR	<i>verb</i>															
VFORM	<i>bse</i>															
PREF-VFORM	<i>active</i>															
SUBJ	<i>&lt; NP : 1 [ human ] &gt;</i>															
COMPS	<i>&lt; NP : 2 [ skill V system V rule ] &gt;</i>															
LEX	+															
SYNSEM   LOCAL	CONTENT	<table border="1"> <tr> <td>RESTR.</td> <td> <table border="1"> <tr><td>RELN</td><td><i>apply</i></td></tr> <tr><td>ARG.1</td><td>1</td></tr> <tr><td>ARG.2</td><td>2</td></tr> </table> </td> </tr> <tr> <td>LEXSEM</td> <td><i>[ ISA use ]</i></td> </tr> </table>	RESTR.	<table border="1"> <tr><td>RELN</td><td><i>apply</i></td></tr> <tr><td>ARG.1</td><td>1</td></tr> <tr><td>ARG.2</td><td>2</td></tr> </table>	RELN	<i>apply</i>	ARG.1	1	ARG.2	2	LEXSEM	<i>[ ISA use ]</i>				
RESTR.	<table border="1"> <tr><td>RELN</td><td><i>apply</i></td></tr> <tr><td>ARG.1</td><td>1</td></tr> <tr><td>ARG.2</td><td>2</td></tr> </table>	RELN	<i>apply</i>	ARG.1	1	ARG.2	2									
RELN	<i>apply</i>															
ARG.1	1															
ARG.2	2															
LEXSEM	<i>[ ISA use ]</i>															
	CONTEXT	<table border="1"> <tr> <td>ACTION-TYPE</td> <td><i>possible likely</i></td> </tr> <tr> <td>U-INDICES</td> <td> <table border="1"> <tr><td>REGISTER</td><td><i>normal</i></td></tr> <tr><td>STYLE</td><td><i>normal</i></td></tr> <tr><td>DIAL-VAR</td><td><i>T</i></td></tr> </table> </td> </tr> </table>	ACTION-TYPE	<i>possible likely</i>	U-INDICES	<table border="1"> <tr><td>REGISTER</td><td><i>normal</i></td></tr> <tr><td>STYLE</td><td><i>normal</i></td></tr> <tr><td>DIAL-VAR</td><td><i>T</i></td></tr> </table>	REGISTER	<i>normal</i>	STYLE	<i>normal</i>	DIAL-VAR	<i>T</i>				
ACTION-TYPE	<i>possible likely</i>															
U-INDICES	<table border="1"> <tr><td>REGISTER</td><td><i>normal</i></td></tr> <tr><td>STYLE</td><td><i>normal</i></td></tr> <tr><td>DIAL-VAR</td><td><i>T</i></td></tr> </table>	REGISTER	<i>normal</i>	STYLE	<i>normal</i>	DIAL-VAR	<i>T</i>									
REGISTER	<i>normal</i>															
STYLE	<i>normal</i>															
DIAL-VAR	<i>T</i>															
DICTCOORD	LEMMA	<i>apply</i>														
	SENSENO	<i>4</i>														
	DICTIONARY	<i>cobuildst</i>														
LEXRULES	3RDSING	<i>applies</i>														
	PRES-PART	<i>applying</i>														
	PAST	<i>applied</i>														
	PAST-PART	<i>applied</i>														

### 1.2.3 The third stage

In attempting to define the most appropriate TFS representation we have been influenced by two factors: the desire to guarantee the usability of our results by a wide range of applications, and the knowledge that this representation would be implemented and tested on the ALEP system. This has meant that, when designing the TFS representation, great care has been taken to reflect the source dictionary data accurately and to conform to the HPSG theoretical framework. The TFS representation format and the related type system is documented in Deliverables 1 and 2; the current system and the procedures which have been developed to convert the information from the Intermediate Format into the TFS formalism will be described and discussed in detail in the Final Project Report.

So far, it has only been possible to test our procedures on the approximately 400 definition statements of the test vocabulary received from Birmingham. In our opinion, without a lot of extra effort, it would be practicable to extend our parser to cover the entire dictionary, or almost. Of course, rules to treat so far unencountered structures would have to be written and implemented,

and it is highly probable that it will be found more convenient to treat a very small number of special cases (the determiners and functional words, for example) manually rather than developing ad hoc procedures. However, this would not effect the general efficiency of the parser. Thus, using our methodology and with not too much extra work, a basic lexicon could be generated, containing a large amount of useful information for NLP applications.

All the procedures have been written in C programming language and run on a SPARC10 workstation.

### 1.3 The Bochum approach

#### 1.3.1 General assumptions.

Unlike other dictionaries, Cobuild exploits the inferential capacity of human language in a strikingly explicit way. The language of its definitions directly reflects a crucial theoretical assumption underlying its design. Consider, for instance, the statement below (which of course analogously holds for all of the remaining definitions in Cobuild),

If you use the word **boy**, you can expect to be presumed to be talking about a male child.  
[HANKS 1987:135]<sup>1</sup>

which nicely encapsulates the legitimate basis for our logical reading<sup>2</sup> of the explanations in the dictionary. Our mapping of

A boy is a male child.

to a logical expression of the (here quantifier-free) expression of BLF<sup>3</sup>

$$\text{boy}.x \rightarrow \text{male}.x \wedge \text{child}.x$$

serves to make this inferential approach in lexicography formally explicit and to incorporate the analysis in an integrative linguistic framework.

Furthermore, our reading of the definitions as conditional statements facilitates the analysis of the dictionary as an ordered self-contained whole and not as a huge disjunction of entries merely ordered on an ergonomic—not theoretical—basis, i.e. alphabetically. The inherent logic of the Cobuild dictionary constitutes a subtle and extremely complex natural order which can now be expressed by the formal means of BLF. Given the adequate representation of such an inherent logical structure, the mapping of Cobuild's natural language definition text to inheritance-based formal linguistic theories is feasible.

HPSG supplies a rich, integrated linguistic apparatus involving multiple inheritance mechanisms which meet the logical requirement of the dictionary. Also, this framework involves a situation semantics component that is in concordance with basic assumptions underlying the Cobuild philosophy and BLF. The relationship between the CONT and CONX features of HPSG reflects the relational theory of meaning in situation semantics. It encodes the correlation between an utterance situation and a described situation, or, viewed from a different perspective, fulfils the implicit lexicographic requirement contained in the following statement:

All statements about word meaning are statements about word use.  
[HANKS 1987:135]

<sup>1</sup> Hanks, P. (1987). "Definitions and explanations." in: Sinclair, J.M. (ed.). *Looking up. An account of the COBUILD Project in lexical computing*. London and Glasgow: Collins ELT. pp. 116–136

<sup>2</sup> Cp. Deliverable 1.

<sup>3</sup> Bochum Logical Form

Finally, the attribute-value system employed for modelling linguistic objects in HPSG makes the theory suitable for at least partial mapping to ALEP syntax. This ensures the applicability of the results in NLP.

To sum up, the methodology of the Bochum team must be evaluated on the basis of its capability to show that there is a systematic relationship between the information in Cobuild definitions and the underlying logic of HPSG lexical entries. The technical feasibility of the resulting mapping process we will try to demonstrate in the next section.

### 1.3.2 The mapping strategy

We distinguish between two levels at which the Birmingham output is mapped to BLF

- the outer level
- the inner level

both of which concern the essential technical aspects of the conversion procedures employed by the Bochum "d21" program. Apart from the specific theoretical assumptions underlying this algorithm, the levels reflect more general tactics of breaking up the informational components of the dictionary definitions. For exemplification, let us now consider the Cobuild entry for "acquire",

1 VB WITH OBJ If you acquire something, you obtain it.

and the corresponding output of the Birmingham parser:

```
( (def_number 241)
  (sense 1)
  (def_type 1)
  (lemma '(acquire acquires acquiring acquired))
  (grammar '(VB with OBJ))
  (pre
    (co-text0
      '()
    )
  )
  (op-word
    (hinge
      '(if)
    )
  )
  (lhs-1
    (co-text1
      (match1
        '(you)
      )
    )
    (head1
      '(acquire)
    )
    (co-text2
      (match2
        '(something)
      )
      '(,)
    )
  )
  (rhs-2
    (match1
      '(you)
    )
    (synonym
      '(obtain)
    )
    (match2
      '(it)
    )
  )
  (post (note '())))
```

The general template for the information content of the "outer level" analysis and representation comprises the storage of the Cobuild grammar, lhs-1, and rhs-2 information in the value ranges of the HPSG features CAT, CONT, and CONX, respectively (See Section 3.2 on the theoretical basis for this correlation). In concordance with common HPSG practice, we decided to fill the PHON value range with lemma information. A full-fledged HPSG analysis would of course entail the encoding of the phonetic information readily available in Cobuild (cp. Section 2.3 on the technical implications). Filling the general template below

$$\left[ \begin{array}{l} \text{PHON} \\ \text{SYNSEM} \mid \text{LOC} \end{array} \left[ \begin{array}{l} \langle \text{"lemma/head1"} \rangle \\ \left[ \begin{array}{l} \text{CAT} \quad \text{"grammar"} \\ \text{CONT} \quad \text{"lhs-1"} \\ \text{CONX} \quad \text{"rhs-2"} \end{array} \right] \end{array} \right]$$

with specific information would, in the case of "acquire", yield the following pseudo-entry:

$$\left[ \begin{array}{l} \text{PHON} \\ \text{SYNSEM} \mid \text{LOC} \end{array} \left[ \begin{array}{l} \langle \text{"acquire"} \rangle \\ \left[ \begin{array}{l} \text{CAT} \quad \text{"VB with OBJ"} \\ \text{CONT} \quad \text{you acquire something} \\ \text{CONX} \quad \text{you obtain it} \end{array} \right] \end{array} \right]$$

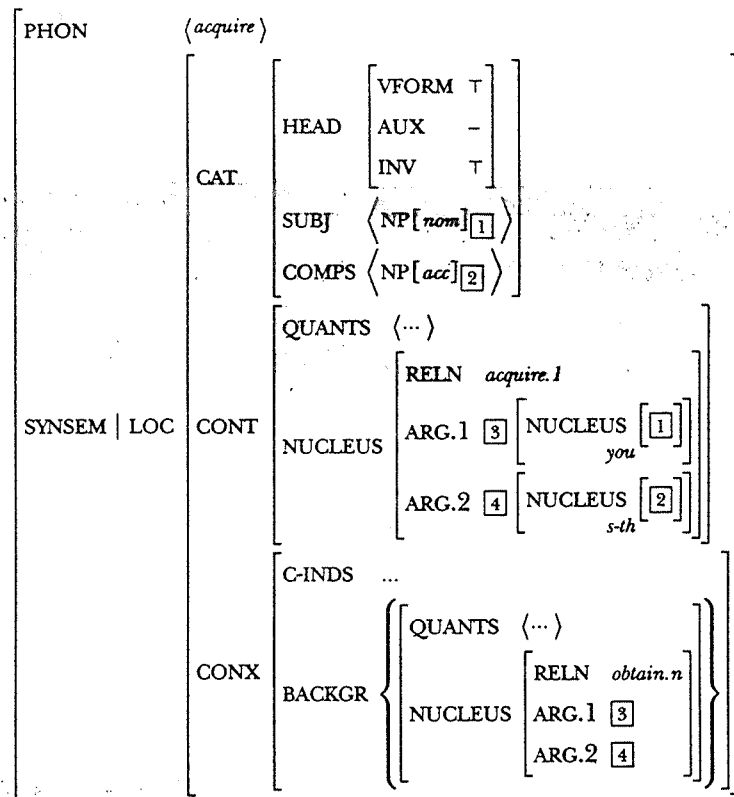
In order to build the remaining information gradually up in more complex feature structures, we now have to proceed not only by further translating the meta expressions of the Birmingham team but also by inserting several feature complexes by default. Consider the general "inner level" template for transitive verbs below:

$$\left[ \begin{array}{l} \text{PHON} \\ \text{SYNSEM} \mid \text{LOC} \end{array} \left[ \begin{array}{l} \langle \text{"lemma/head1"} \rangle \\ \left[ \begin{array}{l} \text{CAT} \left[ \begin{array}{l} \text{HEAD} \quad \text{"VB"} \\ \text{SUBJ} \quad \langle \text{NP}[\text{nom}] \boxed{1} \rangle \\ \text{COMPS} \quad \langle \text{"with OBJ"} \rangle \end{array} \right] \\ \text{CONT} \left[ \begin{array}{l} \text{QUANTS} \quad \langle \dots \rangle \\ \text{NUCLEUS} \left[ \begin{array}{l} \text{RELN} \quad \text{"lemma/head1"} + \text{"sense"} \\ \text{ARG.1} \quad \boxed{3} \left[ \text{NUCLEUS} \quad \boxed{1} \right] \\ \text{ARG.2} \quad \boxed{4} \left[ \text{NUCLEUS} \quad \boxed{2} \right] \end{array} \right] \end{array} \right] \\ \text{CONX} \left[ \begin{array}{l} \text{C-INDS} \quad \dots \\ \text{BACKGR} \left[ \begin{array}{l} \text{QUANTS} \quad \langle \dots \rangle \\ \text{NUCLEUS} \left[ \begin{array}{l} \text{RELN} \quad \text{"synonym"} \\ \text{ARG.3} \quad \text{"match1"} \\ \text{ARG.4} \quad \text{"match2"} \end{array} \right] \end{array} \right] \end{array} \right] \end{array} \right] \end{array} \right]$$

In the value range of the feature CAT—which has not involved a complex value in the "outer level" description—the grammar information supplied by the Birmingham output is broken up into three different HPSG feature specifications: HEAD, SUBJ, and COMPS. The list value of SUBJ contains a

default insertion of a defined HPSG abbreviation reflecting the assumption that transitive verbs lexically require a nominative NP subject syntactically realizing the first semantic argument. The remaining feature paths are filled in a similar fashion.

Finally, by replacing the remaining variables in the general verb template we arrive at the HPSG-entry corresponding to the first sense of "acquire":<sup>4</sup>



<sup>4</sup> The feature RELN will not be employed in the final version of our system. The reason for this move, which is also under discussion in more recent works in the HPSG framework, is mainly due to formal considerations and will be explained in the final report.

## 2 The Users' Perspective

In this section the work done in Project ET-10/51 will be evaluated from the point of view of the users. There are two cross-cutting dimensions:

- a. The type of user. We distinguish three:
  - i. The human being—unassisted,
  - ii. The human being—assisted by machines,
  - iii. The machine—unassisted by human beings
- b. The software package. Because of the design of the Project, we again distinguish three states of software corresponding to our respective laboratories:
  - i. The Dictionary Parser: Birmingham
  - ii. The HPSG Interface: Bochum
  - iii. The Typed Feature Structures: Pisa

Because this is a pilot study, there are many applications that can be proposed but which require the software to be extended to cover more than a sample of the language. Hence, within each category, we shall reflect two stages of the software:

- i. As it is on handover at the end of the Project
- ii. As it might be when extended

### 2.1 Birmingham—The parser in the practice of lexicography

The parser reveals the way in which the meaning of a sentence is organised. In the practice of lexicography it supports and forms the basis for tools for the dictionary compiler and editor.

#### i. *Human being—unassisted*

(n.b. the parser deals with whole-sentence definitions, because it is a partial parser of English. Hence it would have to be adapted for traditional definitions in note form.)

##### • Evaluation of draft definitions:

- Check word class against definition type
- Check all restrictions (cotexts) against established classes of cotexts
- Check superordinate (eventually using draft thesaurus, see below)
- Check discriminators against established classes
- Check matches and results of mismatch procedures

#### ii. *Human being—assisted by machines*

##### • Enhancement of definitions

- Compare definitions and examples for phraseology
- Compare different types of whole sentence definition to generalise parser further
- Initiate systems for sense discrimination
- Initiate systems for drafting definitions from corpus evidence

##### • Review practice

- Explore automatic rewriting procedures to extend range of closed classes in parser
- Study adaptability of phraseology for different purposes—in particular relate whole sentence definition to traditional definitions, by rule if possible

- New tools
  - Draft thesaurus—use thesaurus to improve evaluation of definitions
  - Disambiguate dictionary text
- iii. *The machine*
  - Lexicography—Steps in automating lexicography
    - Drafting definitions
    - Adding new entries, senses to existing dictionaries—revision tools
    - Relating ordinary text sentences to draft dictionary entries
  - Sense disambiguator for ordinary text
  - Preliminaries for open-ended machine-usable lexicon

## 2.2 Pisa—Applying the results of the Pisa syntactic-semantic parser

The objective of the work in Pisa has been to translate and produce instantiations of the syntactically parsed definitions of the Cobuild dictionary provided by Birmingham in a Typed Feature Structure formalism. However, as described in Methodology above, our results have been produced at two different levels: intermediate results; final results in the form of TFS entries. In the following, we will discuss briefly the possible applications of these different results for the three user types recognized in the introduction to this section:

- i. Human user
- ii. Human user—assisted by the machine
- iii. The machine

Obviously, the discussion here below refers entirely to the results that would be obtained once the parser has been applied to the whole dictionary.

### 2.2.1 Intermediate results

At the end of the first stage of analysis, all the information that it has been possible to derive from the definitions is mapped onto an Intermediate Template. For each entry, the IT contains tagged, detailed and explicit orthographic, phonetic, morpho-syntactic, syntactic and semantic information. In particular, information on syntagmatically and paradigmatically related lexical items (i.e. complements and collocates on the one hand, and superordinates, synonyms on the other hand) has been derived by our analyses for each headword. Thus, the IT presents *explicitly* much information which is only contained *implicitly* in the printed dictionary. An example of the format of our results at this stage can be seen in the previous section on Methodology.

#### *Human user*

We feel that there is some scope for an application of these results by the non-computer user. The lexicographer, for instance, might find it convenient to examine the structure of his entries in this format on a printout or the screen. For example, a basic template has been defined for each grammatical category; the results could easily be sorted category by category and compared. In this way, inconsistencies and/or missing information are readily evidenced and can be marked for later correction on the machine. However, we feel that, similarly to other potential users, the lexicographer would be able to exploit the IT results far more usefully with the assistance of the machine.

#### *Human user—assisted by the machine*

User friendly interfaces could be easily implemented to make the information mapped on the IT readily available for different kinds of human users requiring detailed information on a lexical item, its semantic properties and its usage, e.g. translators, language learners, lexicographers again,

etc. Different interfaces can be implemented to meet the needs of different types of users. This would be a very important application of our results: the dictionary user would have direct dynamic access to all the information contained in the lexical entries, wherever it has been stored, and in an interpreted form, instead of being bound by the restrictions imposed by the static alphabetical ordering of the printed volume.

As stated above, we feel that a particularly useful machine-assisted application of the IT results would be in lexicography. It is well known that the current trend for the representation of electronically generated lexical entries is to implement a TFS system. In general, computational linguists are very enthusiastic about the potential of such systems; however, they do not usually meet the same favour from the lexicographer, actually employed in day by day dictionary compilation. In fact, TFS entries are not easy for the human user to handle or analyse. The requirements of the formalism tends to make them "heavy" and they lack flexibility. The lexicographer generally is much more comfortable with a format that is readily interpretable without any need for any specialised training or expertise. We feel that our Intermediate Template meets this demand and would thus be an extremely useful tool in computer assisted lexicography. Furthermore, it can be used to represent information which so far is not handled by standard TFS formalisms. We would have no difficulty in implementing an interface so that the lexicographer could freely query, browse through, compare, merge and extract lexical information from the entries in the IT format.

#### *The machine*

Our Intermediate Template has been designed as the most convenient structure for a first computational model of the lexical information derived from the Cobuild definitions. Our TFS lexicon is then generated automatically by procedures which convert information from the IT into our type system. However, as the IT is entirely theory-neutral, automatic procedures can be developed to extract information and use it in the generation of any kind of computational lexicon.

#### 2.2.2 Final results

An example of how the syntactic and semantic information extracted from the dictionary is represented in our TFS entries can be seen in the previous section.

#### *Human user*

We do not feel that there is a lot of scope for the human user to use our TFS entries in the printed form, especially the non-expert user. However, a printout could be useful for the linguist who wishes to examine the data in detail, perhaps in order to modify or add to the type system implemented.

#### *Human user—assisted by the machine*

On the other hand, we think that our TFS representation could be very useful for different kinds of computer-assisted human users, i.e. any user requiring a consistent formalized representation of the entry. In particular, they could be used by the overall dictionary editor (rather than by the general lexicographer responsible for the compilation of single entries) in a revision of the source or in the production of a new dictionary. The representation of the lexical entries in terms of types enforces a coherent structuring of the entire lexical system, and thus encourages coherence in design of new dictionaries and assists the correction of inconsistencies in the original, by evidencing clearly what is really pertinent in the definition of different kinds of entries.

#### *The machine*

We envisage different types of machine usage:

1. In the construction of computational lexicons for NLP applications: the information contained in our entries should be very useful for systems for the automatic analysis and generation of language. In fact, the information on usage of lexical items and their lexical and syntactic preferences encoded regularly in Cobuild definition statements and represented in our TFS entries is of great importance for NLP lexicons but is not easily derivable in other dictionaries.

2. In sense-disambiguation: we have already tested procedures which can be used for sense disambiguation within the source dictionary, i.e. disambiguation of the superordinates and of lexical preferences (see Deliverable 6, section 5); we have now begun to study the possibility of applying the information on the syntactic and lexical preferences of items for sense disambiguation in texts. For a first discussion, see the section on the Corpus Perspective below.

### 2.3 Bochum—The implications of d2l

#### *Human user—unassisted*

The output of d2l offers a valuable source of information for the theoretical linguist. Given that the definition types analysed in our test vocabulary cover almost the entire set of entries of the dictionary, we can expect our procedures to produce thousands of information-enriched HPSG entries in the near future. Together with a small set of manually compiled entries—the closed class items, like prepositions or pronouns—our system would yield a very large database accessible by either computer or in printed form. In opposition to the ALEP user interface, d2l produces output in a very ergonomic form facilitating easy visualization of information. From a very practical point of view, the HPSG entries produced by d2l can provide numerous examples syntax and semantics studies can be built on—also on a very abstract theoretical level without taking implementational matters into account, if you wish.

The produced entries carry over a wide range of information that has so far not been treated in sufficient depth in the HPSG framework. Our encoding of CONX information, selectional preferences, and hierarchical semantic information in addition to the normally discussed items increases the value of the system as a logically structured and analysed lexicon for the theoretical linguist.

A very important aspect of the results in the form of HPSG entries is the fact that these entries allow a totally different view on grammaticality or well-formedness judgements by theoretical linguists. Due to the corpus-based methods by which Cobuild was produced, the lexical entries derived from it are also enormously meaningful under an empirical point of view. They can actually supply the testing basis and methods that are so frequently demanded by psychologists, who very often blame linguists for their apparently too abstract judgements.

With the logical structure of Cobuild entries isolated, it should cause little effort to make the derived information available also in LFG (Lexical-Functional Grammar) and other logically friendly formats. Also, the phonetic information contained in Cobuild could be supplied for phonological studies. Its extraction from the entries is a trivial task—due to the encoding in the Cobuild data files.

#### *Human user—assisted by machines*

The most important aspect from this perspective seems to us the possible development of lexicographic tools producing input for NLP systems. d2l could very well be used for

- the design of an interface to lexicographic production systems
- the design of a graphical interface in order to facilitate post-processing

The idea is as follows: once the interface to a lexicographic production system has been implemented, each new lexical entry in Cobuild format can be parsed and logically analysed immediately—and automatically—after its definition by a compiler. Since a compiler's intuition should not be hampered by such analytical considerations, an interactive system for post-processing by a theoretical linguist or NLP specialist would ensure the correctness of the results of the logical output. This system must necessarily also have access to corpus information.

#### *The machine*

In section 4.2 we will show that the information encoded in lexical entries for ALEP grammars in current use forms a proper subset of the information that can be supplied by the output of d2l. Therefore it seems reasonable to assume that the majority of problems we encountered in

information extraction from Cobuild would actually become irrelevant if we merely had to derive semantic and syntactic information considered in these systems. For instance, the most problematic part of the algorithm is the analysis of the rhs, i.e. information that is not normally taken into account in existing ALEP systems. Given these minimal requirements for supplying information, the rate of success of our procedures would increase drastically, and hence, the then possible input for ALEP machines would certainly exceed by far what is available now.

Independent of the specific NLP system the output of d2l might be implemented in, the semantic information encoded in our entries is very likely to form the basis of the language-neutral component in automatic multilingual applications or translation systems. Moreover, since the phonetic information supplied by Cobuild entries is easily accessible, it should also cause very little effort to make it available for speech recognition or production systems. Consider for instance an electronic dictionary in which queries can be made acoustically without the use of a keyboard.

## 3 The Theory Perspective

### 3.1 Pisa—What are the theories implied by the Pisa representations?

The task of Pisa (as defined in the Technical Annex) was to extract semantic information from the definitions of the Cobuild Student's Dictionary and to represent the results of this extraction procedure in terms of Typed Feature Structures. The HPSG formalism has been agreed as the Common Interface within the project; HPSG and TFS are fully compatible as the HPSG grammar is based on typed feature structures.

Therefore, Cobuild, TFS and HPSG represent the different theoretical frameworks behind the Pisa lexical representations: the Intermediate Template and the TFS formalisation. Clearly, the two kinds of representation are committed in different ways to the theoretical frameworks above. The Intermediate Template represents the first step in the formalisation of Cobuild entries and for this reason it mainly reflects the Cobuild descriptive framework. It contains all the information clustered around each word sense defined by Cobuild, and there is no commitment to any other linguistic formalism. In this sense, as already mentioned in the previous section, the Intermediate Template can be defined as theory-neutral since the information contained in it could, in principle, be converted into different lexical representation formalisms. In a second stage, this theory-neutral representation has been converted into an HPSG-like representation, using the TFS formalism.

In the following, we illustrate how the different theoretical perspectives have been combined and integrated to produce the Pisa representations of Cobuild lexical entries. Our attention will be mainly focussed on the TFS entries, since they are a result of the convergence of the three different theoretical frameworks adopted by Pisa.

#### 3.1.1 Combining theory and practice

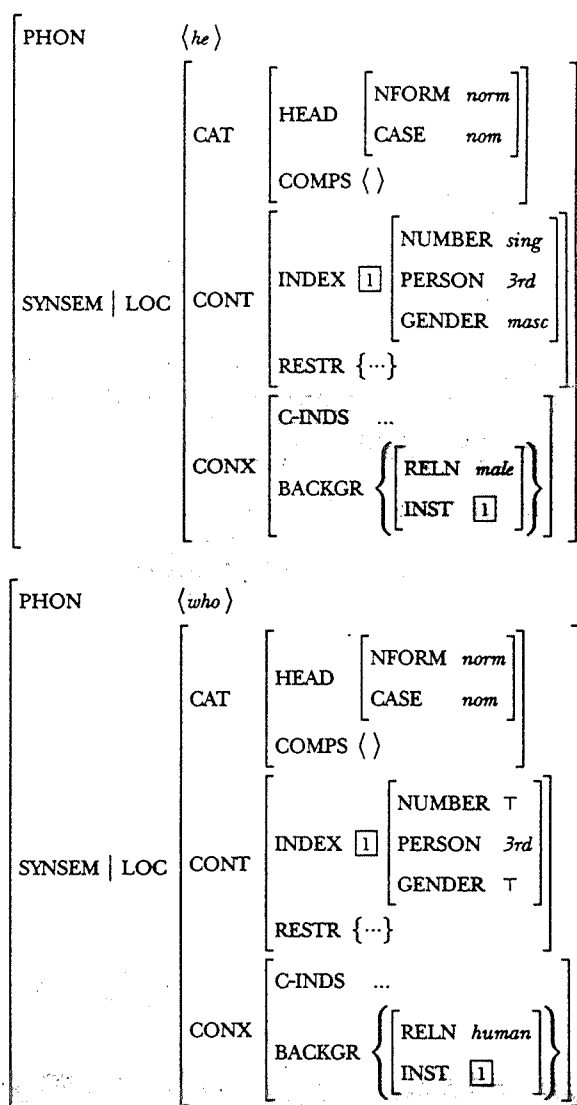
Our current TFS representation can be defined as both theory- and data-driven. It is rigorously data-driven since it is based on the content of Cobuild lexical entries, which in turn reflects the actual usage of words. It can also be defined as theory-driven, since this information on the usage of words has been represented following as far as possible the HPSG specifications. Therefore, the Pisa TFS entries reflect the dictionary data, on the one hand, and should be exploitable by NLP systems inspired by HPSG, on the other. Clearly, the choice of representing information on the actual usage of words—such as typical usages, style, and preferential information on the arguments selected by lexical items—obliged us to revise and integrate the standard HPSG representation in order to include information which is not currently handled by HPSG.

#### 3.1.2 Integrating syntactic and semantic information

In spite of the fact that the main goal of the project was that of extracting and representing semantic information contained in Cobuild lexical entries, we felt that our TFS representation could not be restricted to the representation of semantic information only, regardless of its syntactic implications. In fact, a basic assumption shared by Cobuild and HPSG is that these two levels of linguistic description need to be integrated.

In the Cobuild dictionary, syntactic, lexical, semantic, and pragmatic properties are considered as interacting and contributing to the identification of each word sense. Hence, syntactic and semantic patterns are conceived as strictly interlocked: the distinction of different word senses in the dictionary is determined by the combination of syntactic and semantic factors. Along similar lines, HPSG integrates different linguistic dimensions in its description of linguistic objects: in fact, the description of linguistic signs in HPSG includes specifications of their phonological, syntactic and semantic properties. In particular, the fact that syntactic and semantic aspects are integrated in this description represents one of the main novelties introduced by HPSG.

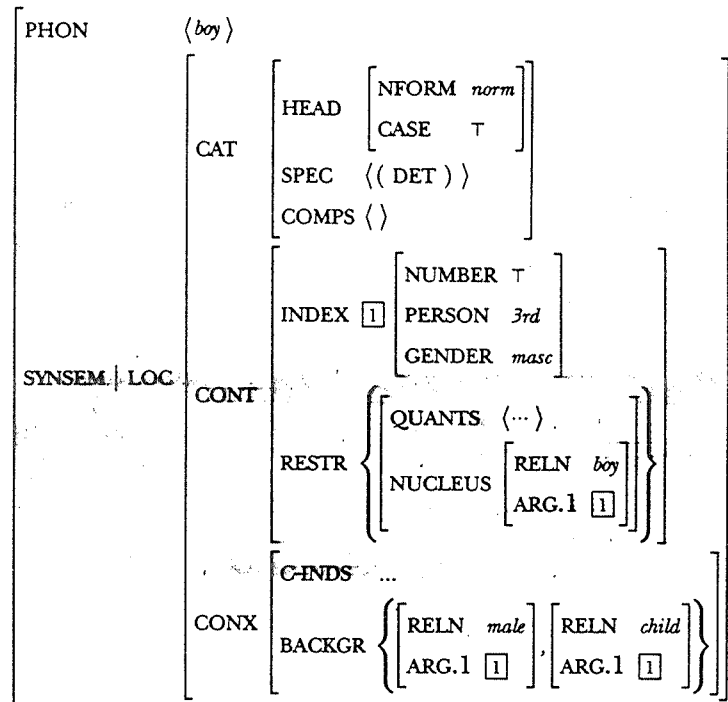
Therefore, the interlocked dependency of syntactic and semantic properties in the description of linguistic objects claimed by both Cobuild and HPSG in the context of different research areas represents the theoretical background supporting our decision to encode both syntactic and



Oversimplifying, one might say that the pronoun “he” actually doesn’t mean anything on its own. Only when you refer to someone by it does it get a certain meaning. That is why the CONT | RESTR range is empty in the HPSG entry. On the other hand, though, you can only refer to someone by “he” when this person is male—and this boils down to *male(x)*—or you’ll deviate from normal usage. This is expressed by the CONX | BACKGR value, which states that the normal usage context or utterance situation always requires a male referent for “he”. Analogously, the same holds for “who”, where normal usage requires a human referent. So far so good, but how can the entry for “boy” tell that the referent of this expression is perfectly suitable for use with both “he” and “who”. Normally, HPSG treatments show little interest as regards the CONX value range of regular nouns, which is hence mostly mentioned only implicitly or ignored altogether. Sadly so, since it also constitutes the much asked-for interface to world knowledge. Apart from that, the CONX value range states a lot of “linguistically relevant” information. And this is, where a dictionary like Cobuild comes in very handy. As pointed out before in Section 1.3.1, it states precisely what we want to have available for an NLP system:

I COUNT N A boy is a male child.

This enables us to encode the predicates *male(x)* and *child(x)* in the CONX range of “boy”—just like before with *male(x)* and *human(x)* in the cases of “he” and “who”, respectively.



Via a grammatical principle which checks the consistency of the CONX values and types the unification of e.g. *male* and *female* would result in a failure,  $\perp$ , or whatever you prefer.

Encoding information of Cobuild in this fashion, however, makes additional revisions of the CONX complex necessary:

- The principle of contextual consistency will have to be formulated more restrictively.
- The CONX range needs to be expanded to convey usage information, which can without any effort be derived from Cobuild entries.

This will only work, however, if the consistency of the types is mentioned explicitly elsewhere, which necessitates statements of type subsumption relations for each entry. In the case of “boy”, we need to state in feature terms (see also below) that

$$boy.x \rightarrow male.x \wedge child.x$$

or that the type *boy* is an extension of both the types *child* and *male*.

Consider the Cobuild entry for “child”: “A **child** is a human being ...”. We get an LHS (or CONT)—i.e. *child(x)*—which implies the RHS (or *conx*)—*human(x)*—resulting in the corresponding type subsumption statement. This guarantees that in the case of “who” and “boy” their respective CONX values (*human* and *child*) unify. Once tied to a “boyish” referent, “who” now carries the CONT semantic information of “boy”, so to speak.

This approach naturally involves a conversion of our predicates to typed feature structures. The correlation is the following:

**First order:**  $boy.1(x)$

**Infonic:**  $\langle\langle boy.1, x, 1 \rangle\rangle$

**FS:**  $\left[ \begin{array}{l} \text{RELN } boy.1 \\ \text{ARG.1 } [x] \end{array} \right]$

**TFS:**  $boy.1 \left[ \text{ARG.1 } [x] \right]$

This allows us to represent the information on “boy” as types in a hierarchy with multiple inheritance:

$$\begin{array}{l} \text{boy.1} \left[ \text{ARG.1 } \boxed{x} \right] \supseteq \text{male.1} \left[ \text{ARG.1 } \boxed{x} \right] \\ \text{boy.1} \left[ \text{ARG.1 } \boxed{x} \right] \supseteq \text{child.1} \left[ \text{ARG.1 } \boxed{x} \right] \end{array}$$

where “ $\supseteq$ ” denotes the dual of the subsumption relation “ $\sqsubseteq$ ”. Hence we assume the following to hold for the lexicon encoded in typed (complex) feature structures:

- it is a weak partial order under the subsumption relation
- it forms a semi-lattice with  $n$  maximal elements, but no greatest
- it forms a hierarchy of semantic types

## 4 The Alep Perspective

### 4.1 The Bochum and Pisa representations of Cobuild lexical entries relative to the Alep formalism

This section reviews the Bochum and Pisa lexical representations with respect to the Advanced Linguistic Engineering Platform (ALEP) formalism and functionality. In particular, their compatibility with ALEP is discussed. As both representations are approximately equivalent in their compatibility with the ALEP formalism, this section reports on those aspects of the approaches which are and are not implementable in ALEP in its current form.

#### 4.1.1 What can be represented in ALEP

In its current form, ALEP can be used to implement a large part of the information present in the Bochum and Pisa entries. The first entry of the Pisa output in Deliverable 7, **abacus**, as well as the first one of the Bochum output, **abdicate**, are referred to below, and will be used to illustrate a number of general features of the representations (both from Pisa and Bochum) and how they can be expressed using ALEP's formalism. **Type and attribute names within the Pisa and Bochum representations will be referred to using capital letters, as with NUMB. Types and attributes defined in the ALEP formalism will be in lower case, between quotes, as 'numb'.**

Starting at the most basic level of the representations, most of the information can be expressed by defining attributes with values. In some cases, the values must be unconstrained, since it would be impossible to specify a comprehensive list of all possible values. This is the case with the specification of the LEMMA attribute, which in the example is **abacus**, but clearly cannot be constrained to some finite list of values, since it could be any one of hundreds of thousands of words in the English language. It would be most logical to constrain other values to a list of possibilities, as is the case with PER or NUMB. The following ALEP declarations can be used for these attributes:

```
lemma    => atom
per      => atom({first,second,third})
numb     => atom({sing,plur})
```

ALEP also provides a method for combining attributes which co-occur often, such as 'per' and 'numb' (agreement), **as well as for specifying boolean operations over the values. This is done by declaring the attributes as being of type boolean, as in the following:**

```
agr      => boolean([first,second,third],[sing,plur])
```

The attribute 'agr' in rules would then allow any one of the values (first&sing), ((second; third)&plur), ( first& sing), etc. The PHON attribute in the example takes a list value, which can be declared as such in ALEP:

```
phon     => list(atom)
```

In many cases, attribute values must be (often quite complex) types, as with CONTENT, which has to allow a number of attributes, viz INDEX, RESTR, LEXSEM. This is declared in ALEP as follows:

```
...
content  => type({content: {}})
...
type(content: {
  index   => type({index: {}}),
  restr   => type({restr: {}}),
```

```
lexsem      => type({lexsem:{}}) },'').
```

Sometimes it is desirable to specify attributes which may not co-occur within the value of other attributes. For example, the value for HEAD (within the Pisa representations) always contains the attribute MAJOR, but NFORM and VFORM cannot co-occur. Similarly, it would not be expected to find COUNT in a verb entry. These restrictions can be implemented in ALEP by declaring the types within a type hierarchy, where the attribute MAJOR is inherited by the subtypes, as follows:

```
type(head:{
  major      => atom({noun,verb,adj}) },'').
```

```
head > { n, v, a } % n, v and a inherit from head
```

```
type(n:{
  nform      => ... ,
  count      => ... },'').
```

```
type(v:{
  vform      => ... ,
  prefvform => ... },'').
```

Given these type declarations and type hierarchy, the types 'n', 'v' and 'a' all inherit the attribute 'major'. An ALEP lexical entry may then contain the information

```
head=>n:{ major=>noun, nform=>norm, count=>count }
```

but may not contain the information

```
head=>v:{ major=>verb, nform=>norm }
```

due to the presence of the 'nform' attribute.

Structure-sharing is used in the entries from Bochum and Pisa, and is implementable in ALEP. In the **abacus** entry, the value of INST is identical with the value of INDEX, and there is a treble occurring **index** in the **abdicate** entry. This is done in ALEP within a given lexical entry (not within the type declarations) by associating a variable with an attribute value, and then using the variable itself as the value of another attribute. Thus an entry could contain the following information:

```
...
index=>VAR=>index:{ per=>third, numb=>sing },
restr=>restr:{ reln=>abacus, inst=>VAR },
...
```

where 'VAR' is the variable used to index the value of the attribute 'index'.

The entries also contain references to other entries in the lexicon. For example, the **abacus** entry specifies information concerning the entry's place in a semantic hierarchy, i.e. ISA within the LEXSEM attribute; similarly, **abdicate** contains a reference to **resign** as the value of RELN.

This could be implemented in ALEP as an attribute with an unconstrained value, where the values **frame** and **resigns**, respectively, are atomic. However, if the entries for **frame** and **resign** actually exist, this type of relationship can be implemented by directly referencing, e.g., the entry for **frame**, or part of it, from within the **abacus** entry. Assuming that an entry is coded as type sign, this type of referencing can be done by the following declaration of the 'lexsem' type:

```
type(lexsem:{
  isa      => list(type({sign:{}})) },'').
```

The value for 'isa' is given as a list, since it may be necessary to specify more than a single dominating entry in the hierarchy. The ALEP entry for *abacus* could then contain the information that an abacus is a type of frame. The reference to *resign* can be done in exactly the same way.

#### 4.1.2 What cannot be represented in ALEP

ALEP in its current state is not well-suited for the representation of all the information present in the Bochum and Pisa representations.

Both Pisa and Bochum representations contain types enclosed in curly braces, indicating that they are unordered set of elements. The ALEP formalism provides no means of representing this information, nor, obviously, can operations over sets be expressed.

Within the Bochum representation for *abdicate*, the types *king* and *queen* form a disjunction. Disjunctions are commonly used by Pisa representations as well. Although disjunction (as well as conjunction and negation) is defined for explicitly specified atomic values via the boolean type (as shown in the previous section), disjunction over other types of values is not defined.

The 'lexical' types, as shown in Bochum representations, are also a problem within ALEP. Taken literally, they are quite infeasible, since they require the declaration of a type for each lexical entry that one wants to use in this way. In order for the system to be able to load a lexical entry properly, the type system itself must already be defined; if a new lexical entry can also be a potential new type, the entire type system would have to be readjusted for each new entry.

#### 4.1.3 Inheritance in ALEP

Inheritance, both within the type system and between lexical entries via a lexical hierarchy (see, for instance, the Pisa contribution to Deliverable 2), is another issue which has been thoroughly discussed within the project.

Within ALEP's type system, it is possible to inherit attribute declarations, i.e. their names and descriptions/constraints. Default values can also be expressed for given attributes, which are then likewise inherited. However, it is a restriction on the assignment of default values that they must be assigned at the point where the attribute in question is declared. This means that one cannot declare an attribute within one type, then assign a value to that attribute within another type that inherits it. This might have been desirable in the example given above, where the type 'n' inherits the attribute 'major'; whenever 'n' is the value of 'head' within some lexical entry, 'major' should have the value 'noun', such that it would be convenient to be able to set `major=>noun` within the type declaration for 'n' and have that attribute-value pair inherited by other types lower in the type system. Another solution is to declare 'major' within the type declaration for 'n' and assign the value there, but this would require redeclaration of 'major' within every subtype of the type 'head'. For example, within the declaration for 'n', you would have a declaration for 'major' as

```
major=>atom({noun}) => noun
```

while, within the declaration for 'v', you would have to have

```
major=>atom({verb}) => verb
```

and so on. This effectively nullifies one of the advantages of the type hierarchy, namely that common attributes need only be declared once.

Multiple and default inheritance cannot be defined within the type system either; as far as the former is concerned, a given type may only have a single super-type.

For the lexical hierarchy, as explained above, ALEP can easily represent the taxonomy by references to entries within other entries. However ALEP has no intrinsic mechanism for actually utilizing these references for inheriting information from superordinate entries.

#### 4.1.4 Summary

To sum up, most of the formal requirements of the Bochum and Pisa lexical representations can be represented in ALEP I.1, including:

- constrained and unconstrained attribute values
- attribute values which are types
- inheritance of attribute declarations
- references to entries from within other entries

The functionality of these features has been tested by the implementation of a few sample entries. ALEP cannot currently be used to represent

- sets (and operations over them)
- attribute value inheritance
- types based on lexical entries
- inheritance between lexical entries

Although standard ALEP is not adequate to represent all of the linguistically interesting information in the Bochum and Pisa representations, the possibility for an open-ended extension of the system functionality can be exploited to implement inheritance between lexical entries.

## 4.2 Reusability of results in existing ALEP grammars

The preceding section has quite clearly shown that the bulk of information extracted from Cobuild entries can be mapped to ALEP syntax—via representation in HPSG—by employing the algorithms of the Pisa and Bochum teams. In cases where the Cobuild information proved not to be representable without substantial effort outside the scope of the project, the problems reside, first, in the lack of formal expressiveness of ALEP, and, secondly, in the developmental status of HPSG, which still has several theoretical shortcomings (cp. Section 3.2 on possible approaches to overcome these).

The relationship between HPSG and ALEP has been discussed in detail in [VAN GENABITH ET AL. 1994]<sup>6</sup>, where we also find the description of the ALEP implementation of a fragment of English. In this section, we will focus on the ALEP type and feature systems employed there and will argue that

- Cobuild information derived by the Pisa and Bochum teams can be mapped to the Essex system straightforwardly.
- A considerable portion of the rich semantic information derived from Cobuild will be lost when mapped to such systems with little semantic expressiveness.

Consider now the Essex-HPSG entry for “eats” from [VAN GENABITH ET AL. 1994:102]:

```
eats~
wordty:{phon=>[phty:{string=>[eats|Rest],restst=>Rest}],
        synsem=>synsemtty:{loc=>locty:{cat=>catty:
                                {head=>mainvty:{vform=>fin},
                                lex=>plus,
                                comps=>[]},
                                %/SLASHED OBJECT
```

<sup>6</sup> van Genabith, J., Markantonatou, S., Sadler, L. & Verhagen, M. (1994). “English HPSG in ALEP.0.” in: Markantonatou, S. & Sadler, L. (Eds.). *Grammatical Formalisms: Issues in Migration*. Luxembourg: Office for Official Publications of the Commission of the European Communities. pp. 91–115

```

        subj=>[synsemty:{loc=>locty:
{cat=>catty:{head=>nounty:{case=>nom}},
con=>nomobjty:{index=>@SUBJ refty:{num=>sg,
pers=>'3'}
}}}],
        con=>psoaty:
{nucleus=>@PSOA psoa2ty:{reln=>eats,
arg1=>SUBJ,
arg2=>OBJ
}},
        nonloc=>nonlocty:
{inher=>udcty:{
%%SLASHED OBJECT
sIash=>[locty:{cat=>catty:{head=>nounty:{case=>acc}},
con=>nomobjty:{index=>@OBJ refty:{}
}}],
que=>[[]]}},
qstore=>quantity:{string=>Quants,restst=>Quants}
}.

```

And now compare and bear in mind the corresponding Cobuild entry:

- VB WITH OR WITHOUT OBJ When you eat something, you put it into your mouth, chew it, and swallow it.

Most of the information expressed in the ALEP entry originates in the complex HPSG templates for a highly economic type system. For instance, the specification of a nominative case NP subject is due to the selectional properties inherent in virtually all English verbs, and the specification of lex as plus is predictable from the absence of phrasal daughters. Information of this kind can be built into general templates a priori and doesn't pose a problem for our mapping algorithms. There is, however, information specific to "eats" and its immediate supertypes that must be encoded in addition to the general information supplied beforehand by theoretical assumptions. Three of these complex informational components are relevant for our present purpose:

- i. the semantic information contained in the cont specification,
- ii. the information on syntactic variation expressed by the nonloc and comps value ranges,
- iii. the semantic information relevant for morphological processes

#### *Semantic information*

The algorithms of both the Pisa and Bochum teams can supply the semantic information needed for the Essex ALEP system without any problem. Since both teams encode the semantic arguments in strict agreement with HPSG terms, this information is in fact already available and only needs to be translated to the ALEP syntax by a trivial process. The ALEP entries in the Essex system do not, however, reflect any semantic restriction or preference information on the subject and object. Also, there is apparently no specification of the CONX feature of standard HPSG or the 'lexsem' feature which are used by the Bochum and Pisa teams, respectively, to represent the superordinate complex of the Cobuild definition. Hence, a very important aspect of the entry for "eat", namely that it inherits information from "take", "chew", and "swallow" cannot be mapped to the present ALEP system although it can be readily supplied by our teams.

#### *Syntactic variance information*

Due to the lexical-semantic emphasis of the project, the teams have so far not studied the encoding of non-local information mainly relevant for syntactic description. For their treatment of unbounded dependency constructions, the Essex team needs to encode slashed categories in the lexical

entries, though. We are convinced that a more thorough study of the grammar information of the Cobuild definitions than was possible in the course of the project would yield the desired results. The extremely large class of English verbs that allow this kind of construction can very probably be isolated by comparatively checking the grammar information of the Cobuild entries and the semantic types of their superordinates and synonyms.

The syntactic information that we can derive and represent immediately is the fact that the syntactic realization of the second semantic argument of "eat" is optional. This follows directly from the grammar information VB WITH OR WITHOUT OBJ contained in the Cobuild entry. Hence, we can ensure the generalization that the transitive and intransitive variants of "eat" share the same semantic arguments—even if the disjunctive COMPS list cannot be represented in ALEP.

*Morphologically relevant semantic information*

For ALEP systems where inflectional information is not handled by lexical rules and where, consequently, inflected and base forms result in separate entries, our teams can easily supply the necessary specifications. Although we have primarily studied the base forms of the Cobuild entries, we can access the relevant inflection information in Cobuild.

In short: both the Pisa and the Bochum teams can supply all the information necessary for HPSG/ALEP entries of the Essex system, but since this information forms a proper subset of information we have available, there is little chance that an ALEP grammar can currently handle all of the linguistically relevant information we can offer.

## 5 The Corpus Perspective

### 5.1 Birmingham—Cobuild definitions and technical vocabulary

For each of the words dealt with below, a concordance listing taken from the ITU corpus was produced, and compared with the definitions in the Student's Dictionary. Two main sources of difference became apparent:

- a. words in the ITU text are generally used in a technical and impersonal context, whereas the definitions in the Student's Dictionary, based on a general corpus, assume a personal subject. A process of 'dehumanisation', referred to in the examples below, would make the dictionary definitions more appropriate in many cases;
- b. in some cases, such as the word 'assembly' given below, the dictionary defines the process associated with the word while the corpus uses it for the product of that process.

Where a definition has been parsed as part of Deliverable 5 or 6 it is asterisked.

#### Assembly

##### Concordance listing

rcular rail supports the mechanical assembly and allows its rotation in azimuth, nna is composed of: -the electrical assembly (as described in 5.2.2) which cons xed-coupling systems Mixed transmit sub-assembly configurations, employing both acing between an analogue multiplex assembly (e.g. a 60 channels FDM supergroup) and monitoring), -compact equipment assembly (e.g. all equipment contained in a upergroup) and a digital multiplex assembly (e.g. two 30 channels PCM groups) - tal networks (ISDNs) Rec. X.3Packet assembly/disassembly facility (PAD) in a pub inal equipment accessing the packet assembly/disassembly facility (PAD) in a pub channels in the baseband multiplex assembly. ii) In consequence, SSB transmissi nsmit) the terrestrial FDM baseband assembly into the minimum number of supergro and since 1986 (CCIR XVIIth Plenary Assembly), it has been studying the performa h-over-elevation (Az-El) mechanical assembly of the wheel and track type. In thi tive elements. It is composed of an assembly of various telecommunication sub-sy n digital. The complete cable is an assembly of multiple individual (10-50) coax -----+ Note 1.- Group: an assembly of 12 telephone channels derived fr kHz). Note 2.- Supergroup (SG): an assembly of 60 telephone channels derived fr be determined by the XVIIth Plenary Assembly of the CCIR, taking 5.2.4 Antenna tion and user data between a packet assembly/disassembly (PAD) facility and a pa ture, thus converting it into a new assembly possessing different statistical an ture, thus converting it into a new assembly possessing different statistical an lack of flexibility in the transmit sub-assembly; -restrictions in the transmit e baseband of a telephony multiplex assembly. The transmitted RF signal is then mmetrical) terrestrial FDM baseband assembly; -the so-called Satellite multiple system which permits the electrical assembly to be steered in any possible orien stal) which supports the electrical assembly (usually on two orthogonal movable

##### Definitions

1

COUNT N

An [assembly] is a large number of people gathered together, especially a group of people who meet regularly to make laws.

2

UNCOUNT N

[Assembly] is the gathering together of people for a particular purpose.

\*3

UNCOUNT N

The [assembly] of a machine or device is the process of fitting its parts together.

Sense 3, the parsed definition, is probably closest because of its reference to 'a machine or device', but it refers to the process rather than the product, an assembly or sub-assembly as a component of a machine, device or system, which is the usage in ITU.

## Message

### Concordance listing

TV carrier (Tx4) in addition to the message carriers (Tx1, 2, 3). If TV transmits satellite channels, an assignment message containing connection information is sent in a way that the uniqueness of a given message is accentuated, thereby allowing a better way that the uniqueness of a given message is accentuated, thereby allowing a better way to transmit the erroneous part of the message. It is clear that the necessity of redundancy is in accordance with the assignment message sent from a transmit station. The in Modulation, FM-FDM-FDMA for message: Similar to Standard A Similar to Standard B

dundancy into the message word. The message source delivers information bits at redundancy into the message word. The message source delivers information bits at redundancy with access but with control of the message source and more specifically with redundancy with access but with control of the message source and more specifically with redundancy more than 20% by removing from the message the time slots corresponding to the more than 20% by removing from the message the time slots corresponding to the redundancy of an observer) from the transmitted message. The characterization of these types of an observer) from the transmitted message. The characterization of these types of redundancy (bits/sample), the DCME will send a message to the telephone exchange (generally used which, for each elementary message to be transmitted, transmits a code used which, for each elementary message to be transmitted, transmits a code redundancy could normally be employed where the message traffic is simple and consists mainly of a PCM system designed for analogue message transmission is readily adapted to redundancy message transmission in

pecially intended for high capacity message transmission<sup>4</sup>. Their main features include the introduction of redundancy into the message word. The message source delivers in the introduction of redundancy into the message word. The message source delivers in

8 Modulation,	FM-FDM-FDMA for message(1):	- One Tx chain
mission	FM-FDM-FDMA for message(1):	- One Tx chain (m
	message:	
Modulation,	FM-FDM-FDMA for message(1):	Similar to St
	FM-SCPC-FDMA for message:	Similar to Sta
	DSI-PSK-TDM-TDMA for message(1)	Transponder hopp

### Definitions

1

COUNT N

A [message] is a piece of information or a request that you send to someone or leave for them.

2

COUNT N with SUPP

A [message] is also the idea that someone tries to communicate to people, for example in a play or a speech.

Sense 1, suitably dehumanised, is closest.

## Power

### Concordance listing

Because of the large number of citations for 'power' in the ITU text a random sample of about 1 in 10 has been given.

ter-facility link 3 dB couplers (low power) (A L'ITALIENNE) 5.4.8.4 Comparison of output power of earth station high power amplifier (see Chapter 5, 5.4.7). The power added in the input combiner of the power amplifier sub-system. Similarly, each effect of an increase in TR. 5.1.1.3 Power amplifiers The order of magnitude of the power in power amplifiers Non-linearity in power amplifiers causes such effects as intermodulation. When the power and spectrum of each intermodulation product is limited by the overall TWT output power and, more precisely, by the output power that can be used for up to about 3 kW output power at 6 GHz. However, at higher frequencies the intermodulation products reduce the satellite average output power by 50% or more to reduce intermodulation products on which links the C/N ratio of the power C of the modulated signal after the transmitter.

possibility of operating with reduced power consumption (for reduced HF power). K itable and can generally be used for power consumption not exceeding 500 W. 5.1. rate, Rb. If S is the desired signal power, Eb the average transmitted energy per the case of satellite transmission, power efficiency is of great importance, low power sub-systems are as follows: -the power entrance facility; including the high (16) The power flux-density (pfd) radiated in a given e to meet the needs of the station's power generating equipment, should be provided current of the power line and on the power generators in the station. 5.7.6 Summary nna system with superposed grids d) Power handling Each generation of satellite IMPATT diode amplifiers with higher power have also been proposed but are scarce. ides total isolation from commercial power interruptions and line disturbances in of operation in that the routing of power may be derived from either or both transmitter measurement A : alarm Po : output power measurement Pr : reflected power measurement e formula for the calculation of the power of each third-order intermodulation product 23 -Intermodulation noise The output power of each carrier divided by the intermodulation possible variations: - of the average power of all the bursts in the TDMA frame (1 oscillator, which has to produce a power of about 10 dBm, may be a conventional e formula for the calculation of the power of each third-order intermodulation product ation HPA and of the required output power of this HPA. The activity factor to be 2.20 gives an example of the output power per carrier versus total input power where: TT/N : test-tone-to-noise power ratio (dB), C/N : carrier-to-noise ratio rna having an effective area Ae, the power reaching the antenna is equal to: W not sufficient to deliver the total power required by all the RF channels. In a tivated and this permits up to a 60% power saving in the satellite transponder (-----) As a general rule, power should be supplied to equipments in the antenna, telecommunication equipment power supply, administration and support services two main sources of power: -the main power supply, with stand-by capability, -the n the correct design of its electric power supply. There are two main sources of mass, the EOL (end-of-life) primary power, the RF (radio frequency) power, the by the payload, -supply of electric power to the payload, -eclipse operation. A d with lightning arrestors. Two main power transformers should be used to step down e receiving amplifiers and with high power transmit amplifiers. They can handle tor and/or heat pipes in the case of high-power TWT; -special heat conditioning g up to 30 W. At 11-12 GHz band, low power TWTAs of between 10 and 20 W and medical and costly than earth terminal power. Typically, one might have to reduce lite mass, BOL(1) power power operation life -----) Maximum transmitted power 34.8 dBW 16

### Definitions

- 1  
UNCOUNT N  
Someone who has [power] has control over people and activities.
- 2  
UNCOUNT N with SUPP  
Your [power] to do something is your ability to do it.
- 3  
COUNT N or UNCOUNT N with SUPP  
If someone in authority has the [power] to do something, they have the legal right to do it.
- \*4  
UNCOUNT N with SUPP  
The [power] of something is its physical strength.
- \*5  
UNCOUNT N  
[Power] is energy obtained, for example, by burning fuel or by using the wind or the sun.
- 6  
UNCOUNT N  
Electricity is often referred to as [power].
- 7  
VB with OBJ  
To [power] a machine means to provide the energy that makes it work.  
  
To [come to power] means to take charge of a country's affairs.  
If someone is [in power], they are in charge of a country's affairs.  
If something is [within] or [in] your [power], you are able to do it.

Senses 4 and 5 convey the idea of a force, and sense 4 brings in the synonym 'strength' which is appropriate in describing the power of a signal. Senses 5 and 6 both cover the idea of power as energy. The concept of capacity which is involved in many of the ITU citations is not covered explicitly.

## Step

### Concordance listing

cs of the signal to be quantized. The step adaptation is then selected in such a nstantaneous value of the quantizing step at time n. This value is adapted as a o-system. The antenna beam is steered step by step so as to obtain a stronger si power transformers should be used to step down the incoming voltage. The transf , it is logical to envisage going one step further and entrusting the satellite ather superfluous manner. The logical step is therefore to design an adaptive co e quantizing, in which the quantizing step is made variable in time as a functio ary techno-economic studies. The first step is to assess the service requirements ather superfluous manner. The logical step is therefore to design an adaptive co oefficients are corrected; -at each step j of the registers, a multiplier calc ry rarely employed. a) Step-track The step-track method uses a so-called climbi osition as shown in Fig. 5.12. If the step-steering of the antenna beam has decre and only a simple beacon receiver and step-track processor are required. Howeve creased the receive signal level, the step-track processor will command the ante ber of possible vectors, some kind of step search algorithm has certain computat ber of possible vectors, some kind of step search algorithm has certain computat nd 5 of Table 5.III), - errors due to step size and to beacon signal level measu e delta modulation (ADM) the variable step size increases during a steep segment ng an adaptive quantizer, wherein the step size is varied automatically in accor . The antenna beam is steered step by step so as to obtain a stronger signal fro = 1 in Table examples). FIGURE 5.12 -Step-track system antenna boresight axis a carrier of a satellite beacon. In the step-track system, no special tracking fee as, especially when combined with the step-track system. A typical block diagram tracking the satellite direction are step-track (Table 5.III - Type No. 4) and beam, is now very rarely employed. a) Step-track The step-track method uses a so ed for wholly terrestrial systems) to step through the international network fro \_ \_ 26.3o - Tracking: monopulse or step track Rec. 465 applies. : in any direction of the- Tracking: step track type Typical environment e partially overcome by combining the step track with a program or memory which (step track) 0.015 0.12 dB (Type No. 5) and 0.015 (r.m.s.) with step tracking (Type No. 4). Note. -The ant - All medium sized Step tracking stems efficient 4 Step tracking age fused disconnect switch feeding a step-down transformer. The transformer sec ntaneous amplitude of the signal. The step variation rule makes use of the multi . If f is the frequency synthesizer step, which should be at least as wide as ay have simple tracking devices (e.g. step-track), while small antennas generall al) is transmitted, the quantizing step will be multiplied by M4 (> 1) at the

### Definitions

1

COUNT N

A [step] is the movement made by lifting your foot and putting it down in a different place.

2

VB with ADJUNCT

If you [step] on something, you put your foot on it.

3

VB with ADJUNCT

If you [step] in a particular direction, you move in that direction.

\*4

COUNT N

A [step] is also one of a series of stages, or a single action taken for a particular purpose.

5

COUNT N

A [step] is also a raised flat surface, often one of a series, on which you put your feet in order to walk up or down to a different level.

If someone tells you to [watch] your [step], they are warning you to be more careful about your behaviour so that you don't get into trouble.

If you do something [step by step], you do it by progressing gradually from one stage to the next.

If a group of people are walking [in step], they are moving their feet forward at exactly the same time as each other.

PHR

step aside

PHR VB

PHR

step back

PHR VB

If you [step back], you think about a situation in a fresh and detached way.

PHR

step down

PHR VB

If you [step down] or [step aside], you resign from an important job or position.

PHR

step in

PHR VB

If you [step in], you start to help in a difficult situation.

PHR

step up

PHR VB

If you [step up] something, you increase it.

Sense 4 seems to match the main usage almost perfectly, although the phrasal use 'step through' is not covered, and the definition given for the phrase 'step down' is not appropriate.

## Address

### Concordance listing

stream of packets each with the same address, and it is the recognition of this address that enables a receiver to select the packet that contains a header containing an address followed by a user data block. A similar case is the CCIR has been meeting since 1983 to address issues related to ISDN performance, and it is the recognition of this address that enables a receiver to select the

### Definitions

1

COUNT N

Your [address] is the number of the house, the name of the street, and the town where you live.

2

VB with OBJ

If a letter [is addressed] to you, your name and address are written on it.

3

COUNT N

An [address] is also a formal speech.

\*4

VB with OBJ

If you [address] a group of people, you give a speech to them.

5

VB with OBJ

To [address] a problem means to deal with it.

Sense 1 is closest to the main usage, but it needs to be 'dehumanised' to take away the 'house' element and replace it with a more general location explanation. Sense 5 matches the single instance of a verbal usage.

## 5.2 Pisa—Testing the Pisa representations on the ITU corpus

In their contribution to this section, the Birmingham group has compared definition statements from the Student's dictionary against concordance listings taken from the ITU corpus. In this part, we will discuss how we could test the TFS representations of the lexical entries, that we have derived from the parsed definitions received from Birmingham, on this corpus. We are, however, restricted in our range of testing as we have just the approximately 400 analysed definitions of the test vocabulary available and in only a very few cases do we have a set of all (or most) of the definitions for the different senses of a single headword. Therefore, at this stage, we are unable to perform an exhaustive testing of our results against the Corpus and our discussion must be, to a large extent, hypothetical.

As was to be expected, a number of difficulties are encountered when attempting to test lexical entries that have been derived from an all-purpose learners' dictionary, constructed on the basis of the evidence provided by a general corpus, against a technical vocabulary. First of all, it should be remembered that the distinction between a general and a technical corpus is not restricted merely to the choice of vocabulary used but includes important differences in the style and structures employed, and these condition considerably the characterization of the lexicon. In the specific case of the ITU corpus we can mention, for example, the considerable use of the impersonal or passive constructions and the frequency of occurrence of nouns used attributively with other nouns, e.g. earth station, message source, telephone channel, etc., etc.. For these reasons, we feel that more valid results could probably be obtained by testing our entries against a general language rather than a technical corpus of this type.

One of the main tenets of the Cobuild theory is that words only acquire sense in context and it is possible to distinguish between the different senses of a lexical item on the basis of its syntactic structure and of the words with which it co-occurs. This is one of the main reasons that led Cobuild lexicographers to systematically specify, for each word sense, the lexical preferences exerted by the lexical item on its arguments, complements or collocates. The evidence provided by the corpus on word usage is thus encoded regularly in the definition statements and represented in our entries. We are convinced that this is very important information for the lexical component of many NLP applications and is needed by systems for both analysis and generation.

Our aim here is to test to what extent this preference information can be used to distinguish between different word senses in the ITU corpus, i.e. between cases of lexical and grammatical homography. (We have already demonstrated how this information could be used for sense disambiguation within the dictionary itself, i.e. on a non-technical type of corpus, on both the paradigmatic and syntagmatic axes, see Deliverable 4, pp. 31–32 and Deliverable 6, pp. 196–199).

In the first part of this discussion, we have concentrated our attention primarily on verbs for two reasons:

1. Much of our initial work on the analysis of the Cobuild definitions focussed on this word class as it was richer in lexical and syntactic preferences, and thus our verb entries tended to be more fully characterized with respect to usage;
2. It is known that in a technical language, it is mainly the nouns that bear the weight of topic specificity, i.e. the technical message; the verbs tend to be of more general sense. Therefore, the nouns in the ITU corpus often have senses which are not represented in the Students' dictionary and/or are used in just one sense; they thus tend to be of little interest from the sense disambiguation viewpoint.

The Cobuild dictionary is human-oriented and very frequently our verbs are characterized as having a preference for a human subject. A technical corpus obviously reveals a different usage; as stated by the Birmingham analysis, the words are generally used in an impersonal context. The first rule to be assumed, therefore, when testing our entries on the corpus was that a +human preference on the subject of a verb was to be ignored.

However, again, one of the problems facing us was that the verbs contained in the ITU corpus and included in our vocabulary subset tended to be used in just one sense, or in a sense not contained in our set of entries.

For instance, if we examine the last example given by Birmingham above for **address**, we find that the dictionary gives three definitions for **address** as a verb:

- 2 ... If a letter is **addressed** to you, your name and address are written on it.
- 4 ... If you **address** a group of people, you give a speech to them.
- 5 ... To **address** a problem means to deal with it.

and only one of them, no. 4, is contained in our test vocabulary. Unfortunately, this is not the sense contained in the corpus listing:

CCIR has been meeting since 1983 to address issues related to ISDN performance

which clearly reflects sense no. 5.

The lexical entries which we would derive from the three verb definitions would each be characterized by a specific preference on the object. (In the case of sense 4, the human preference assigned to the subject would be cancelled by the fact of treating a technical corpus, senses 2 and 5 would not be assigned a subject preference by our procedures.) For convenience, in this hypothetical discussion—hypothetical partly because we have only the parsed definition statement for **address 4**, we refer to the entries as they would appear structured in our Intermediate Template as they are easier to read. Here below, we give first the full entry for **address 4**:

```

def_no           : 337
sense_no        : 4
def_type        : 1
lemma          : address
entry_info      : entry           : address
                 norm_entry      : address
genus_info      : prov_superordinate1 : give a speech
                 syn1            : give a speech
                 genus_prep1     : to
inflection      : address addresses addressing addressed
gram           : VB with OBJ
voice          : active
inference      : possible likely
subj_info      : subj_features1    : human
obj_info       : obj_postmods1     : of people
                 obj_features1    : human-coll

```

We can assume that the **obj\_info** for **address 2** and **address 5** would be as follows:

```

address, 2:
obj_info   : obj_specific      : letter
            obj_features      : -anim, +count

address, 5:
obj_info   : obj_specific      : problem
            obj_features      : -anim, +count

```

The 'letter' of sense 2 and 'problem' of sense 4 would then be referred to the relevant semantic type, once the semantic hierarchy has been constructed from the data (this will be discussed in the Final Report of the project).

Thus 'letter' and 'problem' should here be considered as if they were meta-linguistic terms, representing the set of lexical items (intended as word senses) that will be mapped into the relevant semantic sets. We now show how these representations of the three senses of **address**, VB WITH OBJ, can be matched against the Corpus listing.

In the concordance **address** is found in the same form as in the definition (infinitive) with the object 'issues'. If we look at the entry for **issue** in the dictionary, we find that the appropriate sense is defined as: 'An issue is an important problem or subject that ...' where the superordinate is 'problem|subject'. In our semantic type hierarchy, this sense of **issue** would be linked to the semantic

type of 'problem' and a correct (automatic) sense matching for the corpus example against the lexical entry derived from definition 5 in the dictionary should be possible.

Let us now give another more complete example from the ITU corpus to show how it should be possible to use our entries to help to disambiguate between different senses of a lexical item: both verbs and nouns. The entry for **function** has 5 definitions in the dictionary:

1. COUNT N The **function** of someone or something is their purpose or role.
2. VB If a machine or system **functions**, it works.
3. VB WITH 'as' If someone or something **functions** as a particular thing, they do the work or fulfil the purpose of that thing.
4. COUNT N If one thing is a **function** of another, its amount or nature depends on the other thing.
5. COUNT N A **function** is also a large formal dinner or party.

We have analysed the first three of these definitions and they are represented by our Intermediate Template as follows:

```

def_no          : 11122
sense_no       : 1
def_type       : 3
gram           : COUNT N
lemma         : function
entry_info    : entry           : function
               norm_entry      : function
genus_info    : prov_synonym2   : role
               syn2            : role
               prov_synonym1   : purpose
               syn1            : purpose
inflection    : function functions
art_info      : preferred: def
postcolloc_info : colloc_prep   : preferred: of
               colloc_features2 : human
               colloc_features1 : inanimate

def_no          : 11123
sense_no       : 2
def_type       : 1
lemma         : function
entry_info    : entry           : functions
               norm_entry      : function
genus_info    : prov_synonym1   : works
               syn1            : work
inflection    : function functions functioning functioned
gram          : VB
voice         : active
inference     : possible
subj_info     : subj2           : specific: system
               subj_features2  : inanimate,+count
               subj1           : specific: machine
               subj_features1  : inanimate,+count

def_no          : 11124
sense_no       : 3
def_type       : 1
lemma         : function
entry_info    : entry           : functions
               norm_entry      : function
genus_info    : prov_synonym2   : fulfil the purpose of

```

	syn2	: fulfil the purpose of
	prov_synonym1	: do the work
	syn1	: do the work
inflection	: function functions functioning	
functioned		
gram	: VB with 'as'	
voice	: active	
inference	: possible	
subj_info	: subj_features2	: inanimate
	subj_features1	: human
clause_info	: clause	: as a particular thing

In the corpus, we found 147 listings of the form "function(s)". For space reasons we list just a small subset below:

- 1) satellite is the core of the network and performs the \*function\* of radio-relay in the sky using
- 2) In the international area, global systems such as INTELSAT \*function\* as international common carriers of traffic between major land
- 3) for which the CCIR criteria are given, as a \*function\* of the radio-frequency carrier-to-
- 4) 4 -Antenna gain and 3 dB beamwidth as a \*function\* of antenna diameter for practical antenna efficiencies
- 5) is shown in Fig. 2.6 as a \*function\* of frequency for a distance of 36 000 km
- 6) terms of a change in antenna gain as a \*function\* of the off-boresight angle
- 7) r.p. emitted by the satellite is a \*function\* of the level Cu of the signal received at
- 8) /T)p can be expressed as a \*function\* of separate interference contributions from adjacent frequency satellite channels
- 9) 12 gives the satellite elevation angle and distance as a \*function\* of the earth station's latitude and longitude
- 10) angle R: distance of the satellite as a \*function\* of: - relative station longitude,  
.....
- 13) can be conveniently approximated by a Bessel \*function\* series expansion as: In the above
- 14) approximation, J<sub>1</sub> denotes the first-order Bessel \*function\* of the first kind, by the complex coefficients  
.....
- 22) Digital Circuit Multiplication Equipment (DCME) The \*function\* of the digital circuit multiplication equipment is  
.....
- 80) the multideestination mode. In this case, the multideestination \*function\* is carried out by the DCME. In consequence

As we can see here (Concordances 3–10), and as is confirmed by a complete scanning of the concordances, the vast majority of occurrences of **function** in this technical vocabulary reflect the Cobuild sense 4, which in the dictionary has been indicated as a formal use. It is interesting to note how a sense which is infrequent in a general corpus (Cobuild definitions are ordered by frequency of use) is the most common in a technical corpus. Another quite frequent sense is that attested by Concordances 13 and 14: **function** in the sense of mathematical correspondence; this sense is not given at all in the dictionary. The other senses shown here are sense no. 1 (Concordances 1, 22, 80) and sense no. 3 (Concordance 2). In the entire concordance listing, this is the only occurrence of sense 3 and we found no evidence of senses 2 and 5. While this was to be expected for sense 5, we were a little surprised to find no example of sense 2.

Using the methodology described previously for **address**, it should be possible to use both the syntactic and semantic information we have derived for the different senses of **function** listed above to help us to distinguish between senses 1, 3 and 4 in the concordances.

With reference to Concordance 2, the important clue is given by the grammar code VB WITH 'as' and this should be sufficient to indicate the sense. For the first noun sense, from our analysis we have derived that this sense of **function** preferably takes a definite article, is followed by a prepositional phrase with 'of' + collocate (inanimate|hum), and has as superordinate purpose|role. Without going into a detailed discussion, it can be seen intuitively that the three concordances (1, 22, 80) for this sense respect this pattern.

For dictionary sense 4, we would derive that this sense is followed by a prepositional phrase with 'of' + inanimate collocate. The important aspect of the definition for this sense (If one thing is a function of another, its amount or nature depends on the other thing) is that the two 'things' cited are clearly equivalent and the superordinate amount|nature refers to them and not to the headword **function**. Our analysis would reflect this and indicate that function in this sense is preferably followed by a PP consisting of 'of' + an inanimate item belonging to the 'amount' or 'nature' semantic classes.

This pattern is revealed in all the cases of sense 4 in our concordances. At times the correspondence can be traced directly by following the taxonomic links through the dictionary, e.g:

Concordance 4: diameter -> length -> amount

Concordance 10: latitude | longitude -> distance -> amount

where 'length' is the superordinate for the appropriate sense of 'diameter' in the dictionary, 'amount' is the superordinate for the appropriate sense of 'length', and so on. At times the correspondence will only be evident when a complete semantic type system has been constructed for the lexicon. On the contrary, in our few examples of function, sense 1, when present, the preposition 'of' always governs a concrete item, e.g. station, module, equipment, etc.

To sum up, these two noun senses of **function** (1 and 4) in our concordances are distinguished by both syntactic information (the use of the definite article for sense 1) and semantic information (the presence of a word belonging to the semantic class 'amount' in the PP following function in sense 4).

We think that this example gives a realistic idea of how our lexical entries and, in particular, how the information derived on the syntactic and semantic preferences of a lexical item could be an important part of a set of procedures for text sense disambiguation. Of course, such information is not usually sufficient in itself for sense disambiguation and will have to be combined with other types of syntactic and semantic analyses performed on the text (e.g. POS tagging, listing of possible superordinates for major word classes). It is our intention to continue in the future with experiments in this direction.

### 5.3 Bochum—Cobuild-based entries and ALEP grammars

Consider again a part of the concordance listing for "assembly" presented by the Birmingham team in Section 5.1:

ircular rail supports the mechanical assembly and allows its rotation in azimuth, nna is composed of: -the electrical assembly (as described in 5.2.2) which cons acing between an analogue multiplex assembly (e.g. a 60 channels FDM supergroup) upergroup) and a digital multiplex assembly (e.g. two 30 channels PCM groups) - channels in the baseband multiplex assembly. ii) In consequence, SSB transmissi nsmit) the terrestrial FDM baseband assembly into the minimum number of supergro and since 1986 (CCIR XVIth Plenary Assembly), it has been studying the performa h-over-elevation (Az-El) mechanical assembly of the wheel and track type. In thi tive elements. It is composed of an assembly of various telecommunication sub-sy n digital. The complete cable is an assembly of multiple individual (10-50) coax -----+ Note 1.- Group: an assembly of 12 telephone channels derived fr kHz). Note 2.- Supergroup (SG): an assembly of 60 telephone channels derived fr be determined by the XVIIth Plenary Assembly of the CCIR, taking 5.2.4 Antenna tion and user data between a packet assembly/disassembly (PAD) facility and a pa ture, thus converting it into a new assembly possessing different statistical an mmetrical) terrestrial FDM baseband assembly; -the so-called Satellite multiple system which permits the electrical assembly to be steered in any possible orien stal) which supports the electrical assembly (usually on two orthogonal movable

Assume that this text needs to be translated into German. In order to achieve this via an ALEP system or a similar machine, the text must be parsed by an analysis process employing an English grammar and English lexical entries. The output of the process can then be mapped into a language-independent representation, which then, in turn, serves as the input to a generating system component that produces natural language output in German.

A closer look at the above concordances reveals that the single English expression “assembly” translates into two German words:

- *Konstruktion*, in the cases of
  - electrical assembly
  - mechanical assembly
  - multiplex assembly...
- *Versammlung*, in the case of Plenary Assembly

The third sense of “assembly” in Cobuild would actually translate into *Montage* rather than *Konstruktion* but an insertion of something like “... the result of the process...” would yield a *Konstruktion* reading (Cp. Section 5.1 of the Birmingham team on a general strategy of adapting Cobuild definitions to technical vocabulary in a similar fashion). Sense 1, however matches the necessary reading:

- 1 COUNT N An **assembly** is a large number of people gathered together, especially a group of people who meet regularly to make laws.
- 3 UNCOUNT N The **assembly** of a machine or device is the process of fitting its parts together.

Now, if we encoded the entries in the regular ALEP format with a reduced set of semantic information which mainly serves for syntactic processing, we would arrive at something similar to the following:

```
assembly~
wordty: {phon=>[pty: {string=>[assembly|Rest], restst=>Rest}],
        synsem=>synsemty: {loc=>locty: {cat=>catty:
          {head=>nounty: {nform=>norm},
           lex=>plus,
           comps=>[]},
          ...
```

```
assembly~
wordty: {phon=>[pty: {string=>[assembly|Rest], restst=>Rest}],
        synsem=>synsemty: {loc=>locty: {cat=>catty:
          {head=>nounty: {nform=>norm},
           lex=>plus,
           comps=>[synsemty: {loc=>locty:
            {cat=>catty: {head=>prepty: {pform=>of}}},
            ...
```

This means, for one thing, that the optional complement (a prepositional phrase with “of”) cannot be expressed by a disjunctive COMPS list but must result in two separate entries, as displayed above. But lacking semantic information other than the encoding of the number of arguments and their matching with syntactic categories, these entries would imply correct parses of, for example, “plenary assembly of telephone channels”. Of course, expressions like this are not to be expected in the ITU corpus or other official technical documents, but the question remains into which German term “assembly” must be translated after an English sentence has been parsed.

Unfortunately, there is no simple solution to this dilemma. An ad hoc attempt to supply the necessary information—which is present in the feature structures of the Bochum and Pisa teams—in ALEP entries could involve storing the relevant semantic feature complexes in the respective COMPS lists, like, for example:

```
assembly~
wordty: {phon=> [phty: {string=> [assembly|Rest], restst=>Rest}],
        synsem=> synsemy: {loc=> locty: {cat=> catty:
            {head=> nounty: {nform=> norm},
            lex=> plus,
            comps=> [synsemy: {loc=> locty:
                {cat=> catty: {head=> prepty: {pform=> of}}},
                ...
                con=> psoaty:
                {nucleus=> @PSOA psoalty: {reln=> people,
                arg1=> ...
                ...
```

The MOD\_N feature of adjectives like “plenary” must then also contain this selectional information.

However, this approach would have a number of problematic consequences (Cp. also Section 4.1 on technical aspects of ALEP entries). First, the subcategorisation principle (cp. [VAN GENABITH ET AL. 1994:112f.]) would have to be revised in a rather complex way, and secondly, the semantic information could only be used by rather complicated default techniques, given that inheritance, let alone multiple inheritance, within lexical entries is not possible in ALEP. Encoded as “hard” constraints the semantic information would of course lead to wrong grammaticality judgments otherwise.