EAGLES

Recommendations for the Syntactic Annotation of Corpora

EAGLES DOCUMENT EAG—TCWG—SASG/1.8

Version of 11th March 1996

Contents

1 Contributors

This document, as with all EAGLES documents, is the result of cooperative work involving a number of individuals, contributing in different ways.

1.1 Authors

G. Leech
Department of Linguistics and
Modern English Language
Lancaster University
Lancaster — Tel.: +44.1524.593036
United Kingdom — Fax.: +44.1524.843085
LA1 4YT — E-mail: G.N.Leech@lancaster.ac.uk

R. Barnett
Department of Linguistics and
Modern English Language
Lancaster University
Lancaster — Tel.: +44.1524.65201
United Kingdom — Fax.: +44.1524.843085
LA1 4YT — E-mail: R.Barnett@lancs.ac.uk

P. Kahrel
Department of Linguistics and
Modern English Language
Lancaster University
Lancaster — Tel.: +44.1524.65201
United Kingdom — Fax.: +44.1524.843085
Post Code — E-mail: P.Kahrel@lancs.ac.uk

1.2 Other contributors

H. van Halteren	University of Helsinki
J-M. Langé	IBM Paris
S. Montemagni	ILC–CNR, Pisa
A. Voutilainen	University of Nijmegen

We gratefully acknowledge the help of:
E. Atwell, A. Braasch, N. Oostdijk, N. Ostler, P. Hellwig, M.P. Santalla del Río, G. Rojo

1.3 Editors and Assistants

N. Calzolari
Co-Chief Editor, EAGLES
ILC–CNR
Via della Faggiola 32 Tel.: +39.50.56.04.81
Pisa I-56126 Fax.: +39.50.58.90.55
Italy E-mail: eagles@ilc.pi.cnr.it

J. McNaught
Co-Chief Editor, EAGLES
Centre for Computational Linguistics
UMIST
Manchester Tel.: +44.161.200.3098
United Kingdom Fax.: +44.200.3099
M60 1QD E-mail: jock@ccl.umist.ac.uk

T. Paskiewicz
EAGLES Editorial Assistant
Centre for Computational Linguistics
UMIST
Manchester Tel.: +44.161.200.3097
United Kingdom Fax.: +44.200.3099
M60 1QD E-mail: teresap@ccl.umist.ac.uk

T. Ayazi
EAGLES Secretariat
ILC–CNR
Via della Faggiola 32 Tel.: +39.50.56.04.81
Pisa I-56126 Fax.: +39.50.58.90.55
Italy E-mail: eagles@ilc.pi.cnr.it

A. Enea
EAGLES Webmaster
ILC–CNR
Via della Faggiola 32 Tel.: +39.50.56.04.81
Pisa I-56126 Fax.: +39.50.58.90.55
Italy E-mail: enea@ilc.pi.cnr.it

2 Introduction

This document outlines provisional guidelines for the syntactic annotation of corpora. It consists of five main sections. This section (Section 2) defines the scope of the report: it defines a number of terms (2.1), discusses some problems and issues (2.2), goes into the distinction between the content of syntactic annotation and the method of representation (2.3), and gives a brief overview of existing annotation schemes (2.4). In section 3 a set of annotation levels is outlined and discussed. Section 4 proposes and defines an annotation scheme that can be used to annotate corpora. Section 5 mentions particular problems that may be encountered in the application of any scheme, and section 6 proposes guidelines for the documentation of annotated corpora. In appendix the annotation scheme developed in section 4 is illustrated by applying it to small text samples from different languages.

2.1 Definitions

Syntactic annotation is the practice of adding syntactic information to a corpus by incorporating into the text indicators of syntactic structure: e.g. labelled bracketing, or symbols indicating dependency relations between words. Syntactic annotation is here considered to be separate from morphosyntactic annotation (part-of-speech tagging), which indicates the grammatical class of each word token in the corpus (see EAGLES (1996a) for provisional recommendations on the morphosyntactic annotation of corpora). However, the arguments in favour of this separation are chiefly arguments of convenience and practicality, based on the fact that up to the present, when corpora have been annotated, grammatical tagging has been the first and easiest level of annotation to be added.

It is possible to define syntactic annotation to subsume morphosyntactic annotation. For the sake of consistency, we avoid this definition here[1]. However, as mentioned above, this distinction is an artificial one, and morphosyntactic annotation will be referred to within the study of syntactic annotation, as shown later in the classification of existing schemes. In many cases, morphosyntactic annotation is a prerequisite for syntactic annotation, not an integral part of the parsing process. Syntactic annotation is also considered as separate from semantic annotation, although the boundary between these two is not always clear (See section 3.6 on logical or deep structure annotation.)

The term 'annotation scheme' is used to refer to a specification of a set of annotation practices employed for a particular corpus. For a syntactic annotation scheme, we may also use the term 'parsing scheme' (see further section 5).

We will be concerned only with annotation schemes, not the methods used to apply schemes to a corpus. Annotation schemes may be applied in a number of different ways: manually (as in SUSANNE), automatically (as in the English Constraint Grammar (ENGCG)) or interactively (as in TOSCA). A last possibility is to parse a corpus automatically, manually correcting any errors afterwards (as in the Penn Treebank).

2.2 Problems and issues

Syntactic annotation is more complex and variable than morphosyntactic annotation. In the case of morphosyntactic annotation, the units to be labelled are to a large extent defined in advance (text words being orthographically indicated). In syntactic annotation, not only do we have to determine which labels to apply to segments of the text, but the segments to which they apply have to be chosen from among many possibilities. The way these segments relate to one another also has to be determined. Fortunately, there is considerable consensus about the syntactic segments which have to be recognised in syntactic annotation – e.g. noun phrases and prepositional phrases (there is less consensus about other segments — e.g. verb phrases and subordinate clauses). On the other hand, there is less consensus about how syntactic segments should be defined, as illustrated by the following anecdote (in Sampson 1995:4). During the annual conference of the Association of Computational Linguistics in 1991, NLP researchers from nine institutions were asked to specify the bracketing of a number of example sentences. One of the examples was (1); the brackets in it were the only ones which the nine researchers could agree on.

(1) He said this constituted a [very serious] misuse [of the [Criminal Court] processes].

Thus, the adjective phrase *very serious*, the prepositional phrase *of the Criminal Court processes* and the nominal constituent *Criminal Court* were the only constituents for which consensus could be reached.

Sentences are normally considered to be the maximal units of syntactic analysis or parsing. For our present purpose, the minimal unit is the word. There are therefore two primary points at which syntactic annotation correlates with text: the word and the sentence. In theory, the syntactic definition of these units is independent of the orthographic definition (see Sampson 1995: 153–156). However, in practice the default assumption is that the orthographic word/sentence and the syntactic word/sentence are coextensive[2].

[1] This means that the issue of tokenisation (segmentation of text into words and sentences) is not dealt with in this report. For example, the issues of whether to split off enclitics, and whether to treat multi-word expressions such as compounds as a single token, belong in part to morphosynctactic analysis and in part to text representation. For the same reason, we do not deal here with the representation of merged forms such as French *du* (=*de* + *le*).

[2] In spoken language, 'orthographic sentence' plays no role, sentences being delimited in practice on the basis of syntax and possibly intonation.

2.2.1 Syntactic annotation of corpora, grammars and parsing

The scope of this report is syntactic annotation of corpora. At first glance, a study of such annotation practices is difficult to distinguish from a study of parsers, parsing, grammars, the representation of parses, and the formalisms adopted for such representations. It is clear, however, that we cannot include all of these things in this report. Computational parsing and implementable formal grammars have been the subject of concentrated research over decades. The literature of the field is immense. On the other hand, the syntactic annotation of corpora is a comparatively new field. Apart from work on English, relatively little syntactic annotation of corpora has been carried out on European languages. Clearly, the syntactic annotation of corpora has a close interrelation with parsing (indeed, a major function of a syntactically annotated corpus is to provide a test-bed or a training-bed for wide-coverage parsers). This cannot be ignored in the report: but what we are ultimately interested in is the parsing *schemes* in use to date (i.e. the set of symbols used in the annotation scheme and guidelines for their application), although *how* the corpus is parsed (the parsing *system*) is relevant, albeit indirectly, to our task. For example, robust parsers used for corpus annotation may be rule-based, probabilistic, or may employ a combination of methods. These alternatives may have repercussions in terms of the consistency or reproducibility of the resulting annotation. Thus the system is relevant because there is a relation between the method of parsing, what is actually parsed, and also the nature of the resulting resource.

2.2.2 The size of the corpus and the method of annotation

Another issue is the relation between the size of the corpus and the method of annotation. In order for a corpus to be useful, it needs to be very large, since small corpora do not reveal very much. Although the requisite size of a corpus can to a certain extent be said to be dependent on its use, in general the bigger the corpus is, the better. Increasing corpus size, however, implies automation of annotation. Manual annotation can be slow, expensive and difficult: it needs a number of expert linguists who will need to be trained especially for the annotation task. In theory, the best way to annotate big corpora is by the use of a parser. This, however, is problematic since up to the present there is no reliable parser that can deal with unrestricted text (i.e, a parser that will satisfactorily annotate any text presented to it, be it a newspaper, scientific journals or transcribed conversation or a novel).

There are two intermediate solutions. The first is to use an interactive parsing system. Such a parsing system will parse sentences automatically as far as it is able to, and asks for intervention from a trained operator if it cannot decide on a parse. The TOSCA system (Oostdijk 1991; van Halteren and Oostdijk 1993) is such a system. The second possibility is to let the parser annotate the whole corpus, with manual correction later, if this is deemed necessary and can be achieved. Marcus *et al.* (1993:317) have shown that correcting the Penn Treebank parser's output is significantly faster than manually annotating a corpus: on average, trained operators with a linguistic background needed about 20 minutes to correct 1,000 words; for manual annotating they needed about 44 minutes per 1,000 words. (These figures are useful only for comparison. In absolute terms they do not mean much as the speed of annotation is very much dependent on the complexity of the annotation scheme.)

2.2.3 Horizontal, vertical and other formats

We must, for practical as well as definitional reasons, restrict our attention to corpora considered as collections of texts or textual samples of language. Texts are linear; syntactic structures, on the other hand, are often represented in two-dimensional terms, especially as tree structures, or (in greater detail) as tree structures, the nodes of which are sets of attributes and values. As far as syntactic annotation is concerned, we are interested only in how these two- or multi-dimensioned structures are represented in relation to the linearity of texts.

There are two general commonly-used linear formats for storing, inputting and outputting text data: horizontal and vertical. It is possible to represent a syntactically annotated text in either of these formats, without changing the nature of the annotation. The conversion of a horizontal to a vertical format or vice versa is a relatively trivial operation if undertaken automatically. However, from the user's point of view, the difference between the two formats is certainly not trivial, as it may make the difference between an intelligible and an unintelligible presentation. We will use examples from some corpora to illustrate this.

The first example is from the Associated Press Corpus with Lancaster skeleton parsing annotation[3]. The sentence in (2) can be represented in a horizontal format, as in table 1.

(2) The door, which was equipped with neither bell nor knocker, was blistered and distained.

The labelled bracketed analysis can be represented in a vertical format, as in table 2. The original sentence is in the first column, the part-of-speech tags in the second, and the brackets and labels constituting the syntactic annotation appear in the third column.

Table 3 is an example in horizontal format from the IBM Paris Treebank (Langé 1994).

[3] In a skeleton parse, constituents are identified with brackets, but much detail is omitted: for example, some brackets are left unlabelled, and functional labels such as Subject and Object are not applied.

[N The_AT door_NN1 ,_, [Fr [N which_DDQ N] [V was _VBDZ
equipped_VVN [P with_IW [N neither_LE [bell_NN1 nor_CC
knocker_NN1] N] P] V] Fr] N] ,_,
[V was_VBDZ [blistered_VVN and_CC distained_VVN] V] ._.

Table 1: Horizontal format

The	AT	[N
door	NN1	
,	,	
which	DDQ	[Fr[N]
was	VBDZ	[V
equipped	VVN	
with	IW	[P
neither	LE	[N
bell	NN1	[
nor	CC	
knocker	NN1	]N]P]V]Fr]N]
,	,	
was	VBDZ	[V
blistered	VVN	
and	CC	
distained	VVN	V]
.	.	

Table 2: Vertical format

[N Ce_DDEMMS guide_NCOMS N] [V [P leur_PPCA6MP P] permet_VINIP3
[P de_PREPD [Vi se_PPRE6MP familiariser_VPRN [P avec_PREP
[N les_DARDFP opérations_NCOFP [P de_PREPD [N réseau_NCOMS
[A local_AJQMS A]N]P] [A effectuées_VTRPSFP [P par_PREP
[N les_DARDMP utilisateurs_NCOMP N]P]A]N]P]Vi]P]V] ._.

Table 3: Horizontal format: IBM Paris Treebank

The horizontal format is more compact, and is easier to read so long as the amount of syntactic information interspersed with the words is not too dense. The vertical format is more convenient and more readable if there is too much syntactic information to be conveniently shown in the horizontal format. Moreover, the vertical format lends itself to a number of parallel fields of information, so that (for example) the actual orthographic text (as a sequence of word forms and punctuation marks) can be separated out from the sequence of morphosyntactic tags, and both of these separated from the representation of a phrase structure tree. Other fields may contain corpus location references, and deep syntactic information (such as ellipsis) alongside in a separate field from the surface syntactic information. Table 4 is an example from the SUSANNE corpus (Sampson 1995), which gives an impression of the various aligned information types that can be given. The columns (i.e fields) contain the following information:

Field 1: text references

Field 2: part-of-speech tags

Field 3: the text words

Field 4: base-form (lemmatised forms of Field 3; e.g. *said* is lemmatised as 'say')

Field 5: syntactic annotation (brackets and labels)

The field that indicates the structure of the sentence can be made more graphically explicit by the use of indentation. The example from TOSCA in table 5 illustrates this. On the first level is Utterance, the second level NP, VP and PP, and so on. (This indented format is in fact an intermediate structure, the final output being represented as a tree on the screen.)

2.2.4 Task dependency/reuseability of resources

Corpora, both raw and annotated, may be produced for a wide variety of uses. A corpus may be intended for use purely in linguistic research (studying the variation in language use across genres, or across time; studying the frequency of

A01:0010a	YB	<minbrk>		[Oh.Oh]
A01:0010b	AT	The	the	[O[S[Nns:s.
A01:0010c	NP1s	Fulton	Fulton	[Nns.
A01:0010d	NNL1cb	County	county	.Nns]
A01:0010e	JJ	Grand	grand	.
A01:0010f	NN1c	Jury	jury	.Nns:s]
A01:0010g	VVDv	said	say	[Vd.Vd]
A01:0010h	NPD1	Friday	Friday	[Nns:t.Nns:t]
A01:0010i	AT1	an	an	[Fn:o[Ns:s.
A01:0010j	NN1n	investigation	investigation	.
A01:0020a	IO	of	of	[Po.
A01:0020b	NP1t	Atlanta	Atlanta	[Ns[G[Nns.Nns]
A01:0020c	GG	+<apos>s	–	.G]
A01:0020d	JJ	recent	recent	.
A01:0020e	JJ	primary	primary	.
A01:0020f	NN1n	election	election	.Ns]Po]Ns:s]
A01:0020g	VVDv	produced	produce	[Vd.Vd]
A01:0020h	YIL	<ldquo>	–	.
A01:0020i	ATn	+no	no	[Ns:o.
A01:0020j	NN1u	evidence	evidence	.
A01:0020k	YIR	+<rdquo>	–	.
A01:0020m	CST	that	that	[Fn.
A01:0030a	DDy	any	any	[Np:s.
A01:0030b	NN2	irregularities	irregularity	.Np:s]
A01:0030c	VVDv	took	take	[Vd.Vd]
A01:0030d	NNL1c	place	place	[Ns:o.Ns:o]Fn] Ns:o]Fn:o]S]
A01:0030e	YF	+.	-	.O]

Table 4: Vertical format: SUSANNE

```
-:TXTU()
    UTT:S(act,indic,inter,motr,pres,unm)
        INTOP:AUX(do,indic,pres){Does}
        SU:NP()
            NPHD:PN(pers,sing){he}
        V:VP(act,do,indic,motr)
            MVB:LV(indic,infin,motr){realise}
        OD:CL(act,indic,intens,pres,unm,zsub)
            SU:NP()
                NPHD:PN(pers,sing){he}
            V:VP(act,indic,intens,pres)
                MVB:LV(indic,intens,pres){is}
            CS:AJP(prd)
                AJHD:ADJ(prd){wrong}
    PUNC:PM(qm){?}
```

Table 5: Indented format: TOSCA

particular vocabulary, or structures etc.), or for natural language processing (as a test-bed for an automatic parser; as example text for example based translation, etc.). However, since the production of corpora is so expensive in terms of time and effort, the most desirable corpus would be one that is suited to both ends of the research spectrum. This is not so easy in practice — the aims of these two approaches would be very different, and this would be reflected in the corpus produced, and the annotation applied to it. If a corpus is to be processed automatically, certain phenomena which may be problematic for automatic annotation may be left out, although these phenomena may be more interesting from a linguistic point of view (see further section 5.2.1 on underspecification). With reference more specifically to syntactic annotation, from a NLP perspective, the most important (and difficult) task may be the simple grouping together of certain parts of sentences into constituents, while from the linguist's perspective, this is the simplest (and mainly intuitive) task.

The production of syntactically annotated corpora is a relatively new phenomenon. As more corpora of this type are created, more potential uses will undoubtedly make themselves known. When designing an annotated corpus, it is essential to be aware of this variety of possible uses, since reusability and shareability of such an expensive resource is of great importance. The issue of reusability has recently become a central task in NLP, and therefore the potential multiple exploitations of an annotated corpus should be a primary point in the planning of a corpus annotation project.

2.2.4.1 Collaboration with existing projects

Because of the abovementioned issues of reusability and shareability of NLP resources including annotated corpora, we have endeavoured to make the guidelines presented in this document alignable with the annotation schemes used in relevant EC projects currently being undertaken. In order to achieve compatability of lexica and corpora, we have attempted to ensure that these guidelines comply with those of the EAGLES Lexicon/Syntax SubGroup (for more details see section 4.2 on recommended annotations). The annotation scheme used in the TSNLP Project (Test Suites for Natural Language Processing) has also been considered in the formation of our guidelines.

2.2.5 Syntax/semantics and syntax/discourse boundaries

Although these guidelines deal purely with syntactic annotation of corpora, the problem of the boundary between syntax and semantics necessarily arises. There are cases where a phenomenon clearly falls within a semantic analysis rather than a syntactic one, but there are also cases where the boundary is slightly more fuzzy. For example, it is common practice within most approaches to grammar to subcategorise adverbial clauses as adverbials of e.g. ‘place’, ‘manner’ or ‘time’. These categorisations are primarily semantic — i.e. the choice of category is made dependent largely on semantic information — in this case the meanings of the words in the adverbial clause. However, these semantic categories involve syntactic properties, for example in English whether an adverbial is of ‘place’ or ‘time’ will affect its position in the sentence.

Although a semantic analysis is beyond the scope of the guidelines outlined here, mention will be made of some more semantically-based distinctions. The boundary between syntax and semantics is itself much disputed, and in terms of NLP, it has been shown through work on POS tagging that often semantic information may be an aid to automatic class identification.

2.2.6 Form/function distinction

In defining the labels for units, the distinction between form and function is an important one. In syntactic annotation of corpora, the level of analysis to which we devote most attention in this document relates to *form* rather than *function*. Function can be seen to be a further level of analysis, in which different categories may be shown to fulfill the same syntactic function within a large syntactic unit, e.g. a Noun Phrase may function as an Adverbial adjunct within a sentence, as may also a Prepositional Phrase or an Adverb. This problem compares directly with the issue in morphosyntactic annotation, in which the function of a word may be different from the morphosyntactic class of the word itself. Thus a noun may function as a modifier of another noun (fulfilling an ‘adjectival function’), e.g. *colour* in *colour television*, or a past participle form of a verb may function as an adjective, e.g. *disputed* in *disputed territory*. As a general rule, it is common practice to annotate firstly at a **form** level, and at a functional level optionally afterwards. This can be seen as advantageous since the initial level of annotation is one which provides basic descriptive information about the language (correlating directly with the type of information included in a lexicon of the language), the further annotation at a functional level would provide a richer, deeper analysis.

2.3 Text representation and the representation of annotation

It is important to draw the distinction between the content of syntactic annotation, and the method of representation. In this report we are concerned only with the former. The Text Representation subgroup of the EAGLES TCWG will be concerned with not only the representation of orthographically encoded information, but also with how to represent linguistic annotations, including syntactic information of various kinds. It will base these representations on the guidelines of the Text Encoding Initiative (Sperberg-McQueen & Burnard (eds) 1994), which are designed for the interchange of documents or texts in circumstances where common standards are required for corpus development activities. For the purposes of this report, we cannot prejudge the forthcoming work of the Text Representation subgroup in this area, and therefore we will represent syntactic annotation here in the format used for that purpose by those who devised the annotation schemes we are exemplifying.

2.4 Existing schemes

Although we have endeavoured to take account of all available work on syntactic annotation, it is regrettable that much of the work undertaken to date has been on English. The extension of such work to other languages is relatively new, and as shown below, tends to be an extension of existing schemes developed for English, as instantiated in corpora or corpus parsing systems. The list below gives a brief overview of existing schemes; for a complete overview, see EAGLES (1996b) and EAGLES (1996c). In the list, schemes/systems which are currently under development (as at May 1995) and for which no data are available are shown in square brackets[4].

English: Lancaster-Leeds Treebank; Lancaster Parsed Corpus; SUSANNE; Penn Treebank; ENGCG; TOSCA; CCPP; POW Corpus[5].

Dutch: AMAZON/CASUS system in preparation (Nijmegen).

German: [Helsinki Constraint Grammar]

Finnish: [Helsinki Constraint Grammar]

Danish: Extension of the Helsinki Constraint Grammar, at the Institute of Linguistics, University of Aarhus

Swedish: [Helsinki Constraint Grammar]

French: IBM Paris (parallel to Lancaster/IBM work); [Helsinki Constraint Grammar]

Spanish: [extension of Penn Treebank scheme]; preliminary work is being carried out at the University of Santiago de Compostela under Guillermo Rojo

Italian: no annotated corpora as yet, but work is being carried out at the Institute of Computational Linguistics in Pisa

It follows that what is proposed in this report is based on the limited experience of working with a small number of languages, and can be only tentatively applied to European languages in general.

[4] Currently the Helsinki Constraint Grammar is also being extended to Basque, but since Basque is not an official EU language, it will not be covered in this report. However, as Basque is typologically distinct from the Indo-European languages with which the EAGLES guidelines are initially largely concerned, efforts at parsing it may produce interesting results.

[5] Some of these names refer to the parsing system employed and others to the resulting treebank or syntactically annotated corpus.

3 Layers of annotation

Turning to the content of syntactic annotations, we need to recognise different layers of information which may or may not be encoded in a particular syntactic annotation scheme:

a. Bracketing of segments;

b. Labelling of segments;

c. Showing dependency relations;

d. Indicating functional labels;

e. Marking subclassification of syntactic segments;

f. Deep or 'logical' information;

g. Information about the rank of a syntactic unit;

h. Special syntactic characteristics of spoken language.

3.1 Bracketing of segments: layer (a)

Bracketing of segments involves the delimitation by bracketing, usually with square brackets, of the segments which are recognised as having a syntactic integrity (sentences, clauses, phrases, words). In example (3),

(3) [[He] [walked [into [the garden]]]]

the garden is one unit within *into the garden*, which in turn is a unit within *walked into the garden*. *He* is a unit in its own right. The outermost brackets indicate that the whole sentence is a unit.

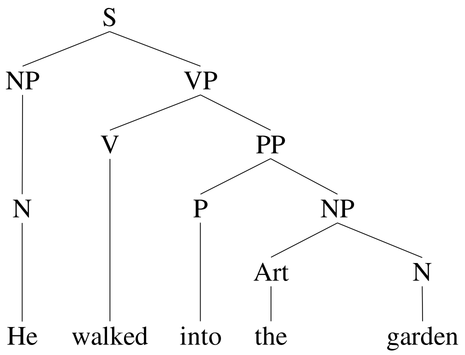
3.2 Labelling the category of segments: layer (b)

This layer of information is the indication of the formal category of the non-terminal syntactic units or constituents identified by bracketing, such as Noun Phrase, Verb Phrase, Relative Clause. For example, adding labels to the above string yields the labelled analysis of (4):

(4) [S [NP He NP] [VP walked [PP into [NP the garden NP] PP] VP] S]

Such bracketed strings can be represented as a tree structure, as in (5) (for completeness' sake the part-of-speech labels N, V, P, and Det have been added).

(5)



3.3 Showing dependency relations: layer (c)

This layer of information comprises head–dependent relations between words, e.g. adjectives and the nouns they modify. It is often assumed that the showing of dependency relations between words (dependency syntax) is an alternative to labelled bracketing (phrase structure syntax) as a way of indicating syntactic constituency relations, and indeed the two methods of indicating syntactic structure have overlapping functions. But they also convey different kinds of information (see below), and since both may be (independently or simultaneously) marked in a text, there is no difficulty in treating them as separate layers of annotation for the present purpose.

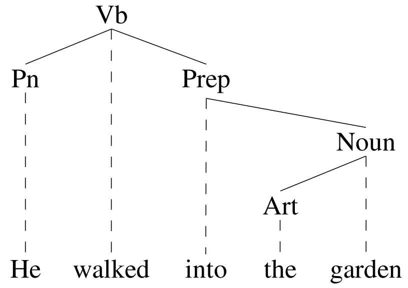
Dependency relations are traditionally indicated with arrows; these point from a head to a dependent, as in (6), although they may point in the opposite direction, showing the relation of a dependent to the head, as in (7). Note that there are some differences of opinion as to which words are heads and which words dependents. For example, within the NP, the noun may be regarded as the head of the determiner, or vice versa.

(6) He walked into the garden

(7) He walked into the garden

In many cases, dependency relations can also be represented by a tree structure, as in (8)[6]. Heads are shown as nodes at the upper ends of branches (slanted lines) and dependents are shown at the lower end of branches. Thus, *walked* is analysed as the head of both *he* and *into*; *into* of *garden*; and *garden* of *the*.

(8)



An important difference between phrase structure and dependency trees is that in phrase structure trees the segments related to one another may or may not correspond to single words, whereas in dependency representations the segments related to one another are words.

As far as we know, the ENGCG parser is the only system of corpus annotation that uses a partial dependency syntax, and can be used as an exemplar[7]. An example of ENGCG output is given in table 6; the output was obtained from LingSoft's e-mail parser. Instead of the arrows of the traditional dependency representation, the symbols < and > are used to indicate dependents; < means that the word's head precedes, > that it follows. The link is established in the syntactic function label. For example, @DN> indicates that *an* is a determiner (D) whose head is a noun (N) that follows. If the head is not the next noun but the second following one, the notation @DN2> is used. The symbols @+FMAINV and @-FMAINV represent a finite verb and a non-finite verb respectively.

Although there are a number of resemblances between phrase structure and dependency trees, it is not always easy or possible to convert one to the other. In the classical phrase structure representation, as in (5), heads are not marked at all, so a mechanical conversion will require stipulation of what the head of each dependent is in order to be able to create the dependency chain. Conversely, since in classical dependency syntax constituency plays no role (constituents as such are not indicated), the boundaries of a constituent have to be reconstructed from a subtree of a dependency tree. Thus, in (8), the part of the dependency tree dominated by Prep defines a prepositional phrase. However, dependencies may also be discontinuous, not mapping on to a context-free phrase structure tree.

Since phrase structure trees and dependency trees convey overlapping types of information, it is unusual for a syntactic annotation scheme to include both. We regard them in this report as alternative methods of indicating structural relations between parts of a sentence.

3.4 Indicating syntactic function labels: layer (d)

Segments can be labelled according to their syntactic function, such as Subject, Object, Adjunct,. . . . Examples of this layer of information are found in the SUSANNE and TOSCA phrase structure annotations in tables 4 and 5 and in the ENGCG dependency annotation in table 6. It shows that syntactic functions, together with dependency relations, may be annotated without bracketing, which means that the functions are assigned to words rather than to constituents. The @-symbol indicates a syntactic label (e.g. @OBJ designates Object). The arrows > and < indicate dependency relations; for details see above.

Indicating syntactic functions overlaps with (c) — dependency analysis — to some extent. For example, the dependent–head relation between adjective and noun in *wild flowers* may be alternatively considered a matter of assigning syntactic function labels to *wild* (modifier) and *flowers* (head). However, these two types of annotation would be shown to be different if we took an example of a noun followed by a prepositional phrase: *bottles of wine*. Here, in a functional labelling of constituents, the whole prepositional phrase *of wine* would be a modifier of *bottles*; whereas in a dependency analysis, the word *of* would be the dependent of the head *bottles*, and the noun *wine* in turn would be a dependent of *of*.

3.5 Marking subclassification of syntactic segments: layer (e)

This involves assigning feature values to phrases or words, e.g. marking a Noun Phrase as singular, or a Verb Phrase as past tense. A feature-based syntax has been modelled by the Text Encoding Initiative (Sperberg-McQueen & Burnard (eds) 1994), in which syntactic information may be represented by means of attribute–value pairs. If necessary, this kind of information can be added in a compact way by adding various subscripts to syntactic segments. Table 7 is an

[6] But not always: note that a dependency analysis may have crossing branches, or tokens which are dependents of more than one head.

[7] In the sense that ENGCG is the only wide coverage parser. Other parsers using a dependency syntax exist. See references in Fraser & Hudson (1992).

```
”<*i>”
        ”i” <*> <NonMod> PRON PERS NOM SG1 SUBJ @SUBJ
”<started>”
        ”start” <SV> <SVO> <P/on> V PAST VFIN @+FMAINV
”<work>”
        ”work” N NOM SG @OBJ
”<on>”
        ”on” PREP @ADVL
”<an>”
        ”an” <Indef> DET CENTRAL ART SG @DN>
”<*english>”
        ”english” <*> <Nominal> A ABS @AN>
”<description>”
        ”description” N NOM SG @<P
”<within>”
        ”within” PREP @<NOM @ADVL
”<the>”
        ”the” <Def> DET CENTRAL ART SG/PL @DN>
”<*constraint>”
        ”constraint” <*> N NOM SG @NN>
”<*grammar>”
        ”grammar” <*> N NOM SG @NN>
”<framework>”
        ”framework” N NOM SG @<P
”<proposed>”
        ”propose” <Vcog> <SVO> <SV> PCP2 @<NOM-FMAINV
”<by>”
        ”by” PREP @ADVL
”<*karlsson>”
        ”karlsson” <*> <Proper> N NOM SG @<P
”<$[>”
”<1990>”
        ”1990” <1900> NUM CARD @ADVL
”<$;>”
”<1994a>”
        ”1994a” <1994a> NUM CARD @ADVL
```

Table 6: ENGCG output

example from the SUSANNE Corpus (showing only part of a sentence), with, in the right hand column, a singular (proper) noun phrase (Nns), and past tense verb phrase (Vd).

...				
A01:0010b	AT	The	the	[O[S[Nns:s.
A01:0010c	NP1s	Fulton	Fulton	[Nns.
A01:0010d	NNL1cb	County	county	.Nns]
A01:0010e	JJ	Grand	grand	.
A01:0010f	NN1c	Jury	jury	.Nns:s]
A01:0010g	VVDv	said	say	[Vd.Vd]
...				

Table 7: Marking subclassification in SUSANNE

Similarly, the ENGCG annotation scheme gives subcategorisation information, specifying values such as PERS (personal pronoun), NOM (nominative) and SG1 (first person singular), and for verbs <SV> (intransitive), <SVO> (monotransitive) and <SVOO> (ditransitive) (see table 6 for examples). Subclassification is also coded in TOSCA (see table 5).

3.6 Logical relations of various kinds: layer (f)

This includes a variety of syntactic phenomena, such as co-referentiality (in e.g. control structures), cross-reference (or substitution), ellipsis, control, traces and syntactic discontinuity. In table 8, an example from the Penn Treebank, co-referentiality is marked by indices[8]. The subject position of the infinitive phrase *to go* has the same index — 8 — as the subject of the conditional clause (NP-SBJ-8 *you*).

```
(S (NP-SBJ You)
    (VP can
        (VP leave
            (SBAR-ADV if
                (S (NP-SBJ-8 you)
                    (ADVP really)
                    (VP want
                        (S (NP-SBJ *-8)
                            (VP to
                                (VP go)))))))))
```

Table 8: Marking cross-reference in the Penn Treebank

3.7 Information about the rank of a syntactic unit: layer (g)

From the schemes looked at so far, this has not been marked explicitly, although such information is obtainable from most parser outputs by the embedding of marked brackets. In some systems (e.g. TOSCA) this is shown by indentation. In table 5 there is a number of different ranks, such as UTT (utterance); SU, V and OD (Subject, Verb and Direct Object); NPHD — noun phrase head, MVB — main verb, CS — subject complement, . . . [9].
Within some annotation schemes, the concept of rank is applied to general categories of constituents, words being of lower rank than phrases, phrases being of lower rank than clauses, and clauses being of lower rank than sentences. These ranks are found, for example, in the SUSANNE annotation scheme and are used as a basis for deciding whether to represent one-word constituents in the labelled bracketing notation (Sampson 1995: 172f).

3.8 Information about spoken language non-fluency phenomena: layer (h)

Spoken language corpora show a range of phenomena that do not normally occur in written language corpora, such as blends, false starts, reiterations and filled pauses. In syntactic annotation, it has to be decided whether to include such phenomena in a parse tree, and, if so, how. To the best of our knowledge this has been implemented only in the Polytechnic of Wales (POW) corpus (Souter 1989) and in the International Corpus of English. Sampson (1995: Ch.6) mentions its possible inclusion in a future release of the SUSANNE Corpus, and on-going work at Lancaster on the British National Corpus will include incorporation of non-fluency phenomena in the skeleton parsing of both written and spoken data. There is also a proposal to undertake an analysis of such phenomena in the parsing of the British component of the International Corpus of English (Aarts 1992; Greenbaum 1992).

3.9 Conclusion

Layers (a)–(h) represent a loose hierarchy of informational details, and in some cases a 'higher' type of information cannot be added unless a 'lower' (i.e. more basic) type has already been provided: e.g. (a) is a pre-requisite for (b). Table 9 below summarises which of the features distinguished in (a)–(h) have been implemented in a number of annotation schemes. (It is a 'quick overview'; a more detailed overview is given in the documents SASG2 and SASG3.) However, there is no strict precedence between the levels, and many combinations of information types, other than those represented in the table, are in principle possible. It is conceivable, for example, that someone might wish to indicate functional information about syntactic segments in a corpus while ignoring other kinds of information.

[8] These logical relations are marked only in the second phase of the Penn Treebank Project.
[9] Discourse notions like 'utterance' will not be dealt with in this report, although they are sometimes used to mark the outer bounds of parse trees.

	Lancaster	SUSANNE	Penn Treebank	IBM Paris	TOSCA	ENGCG	POW	CCPP	AMAZON/CASUS
Bracketing (a)	x	x	x	x	x	–	x	x	x
Labelling (b)	x	x	x	x	x	–	x	x	x
Dependency relations (c)	–	x*	x*	–	–	x	x	–	–
Syntactic functions (d)	–	x	x	(x)	x	x**	x	x	x*
Subclassification (e)	–	x	–	–	x	x*	–	–	x
Logical information (f)	–	x	x	–	–	x*	x	–	–
Rank (g)	x	x	x	–	x	–	x	x	–
Spoken language (h)	–	x*	–	–	x*	–	x	x	x*

* intended for future release
** at word level
– not implemented

Table 9: Overview of annotation levels in selected corpora

4 Guidelines

The notion of standardisation of syntactic annotation of corpora is a problematic one. Even more than in the case of morphosyntactic annotation, the situation with respect to syntactic annotation is variable and fluid. Certain practices have become fairly widespread (e.g. common treebank conventions adopted in the Lancaster, Penn Treebank, and SUSANNE treebanks for English, and at IBM for French. Similarly, the formalism of the Helsinki Constraint Grammar, which has been applied to English, is currently being extended to Finnish, German, Swedish, French and Basque). But it would be premature to impose a standard on all syntactic annotation activities under the aegis of the EC. Work in this area is still at an early stage. A scheme chosen for syntactic annotation may have a marked effect on the success of a parser working with that scheme. Since no completely successful parser has as yet been developed, the imposition of any particular scheme could be detrimental to future research. The guidelines outlined here are therefore only intended to be preliminary. The scheme adopted for syntactically annotating any corpus is one which is partly task dependent (see section 2.2.4). The framework presented here is therefore a basic introduction to the task, outlining some canonical categories, and some of the problems and difficulties that may arise from their application. There are many choices to be made in the analysis of natural language, and the preference of one approach to another will depend in part upon the intended purpose of the corpus.
Following a similar procedure adopted in the EAGLES Guidelines guidelines for morpho-syntactic annotation of corpora (EAGLES (1996a)), the information types given in the guidelines are divided into three sections, of which the first is empty in this case:

1. **Obligatory annotations**: required if the parsing scheme is to be conformant with EAGLES standards.
2. **Recommended annotations**: not required, but recommended as a 'default' in the sense that such information should not be omitted unless there are good reasons for doing so.
3. **Optional annotations**: where inclusion or omission is not required or recommended, but is a matter to be justified in its own terms, taking account of such factors as practical constraints, the task orientation, or the language of the text corpus.

4.1 Obligatory annotations

Because of the variant nature of syntactic annotation, and the almost infinite combinatorial possibilities, it is suggested that no part of the syntactic annotation be regarded as obligatory. The first layer (a) in the 'hierarchy' of annotation, bracketing (see section 3.1), could be seen to be a possibly obligatory level, and indeed for a constituent analysis, it would be. However, as we have seen, there are schemes in existence already that do not actually group together the words making up the constituents (ENGCG), and although this can produce a rather shallow analysis, it must still undoubtedly be regarded as a form of syntactic annotation.
As mentioned earlier, syntactic annotation may be undertaken for a variety of reasons, and therefore to impose an obligatory level of annotation would be to assume that there is a level of analysis common to all of these aims. This is not

the case — for some purposes (e.g. some kinds of parser testing), only bracketing, and no labelling may be necessary. For others the labelling of certain constituents, e.g. noun phrases, may be required, while the rest of the sentence may be left unannotated. For all these reasons, the obligatory level of annotation is to be left empty.

4.2 Recommended annotations

In this section, we propose informal criteria for the marking of a number of labels, corresponding to the category labels proposed by the Lexicon/Syntax SubGroup. The main motivation for following the recommendations formulated by the Lexicon/Syntax Subgroup is that the structure of sentences is determined to a large extent by the lexical properties of words, and in order to ensure compatability of lexicons and corpora, if possible the syntactic labellings employed in lexicons should be employed in the annotation of corpora.

The recommended syntactic labels and constituents proposed in the forthcoming report of the Lexicon/Syntax Subgroup are widely recognised and well-established in current corpus annotation practices that use phrase-structure-like representations. They can be seen as 'familiar landmarks' which anyone wishing to make use of a syntactically annotated corpus would probably expect to find. Arguably, they represent the constituents and classifications that are to be used to arrive at a minimally interesting annotated corpus; these constituents are outlined and defined in this section.

Apart from the recommended syntactic categories, we define and illustrate a number of optional ones. These optional categories are merely illustrative for the purposes of the present report, and represent a cross-section of the annotation practices currently employed. They will be outlined and defined in section 4.3.

The form of the codings we use is tentative and has been chosen for convenience (see text representationsection 2.3). We use rather verbose labels for readability's sake. The exact notation, e.g. whether to use a notation like Subj-Agent or Su/Ag to denote that a constituent is the Subject of the clause and has the semantic function Agent, will presumably be determined in the documents to be produced by the Text Representation Subgroup within EAGLES. Essentially, the mark-up of the codes is trivial, since (assuming that the labels have been applied consistently and unambiguously) they may be mapped onto any other mark-up.

Each category will be informally defined and illustrated by one or more examples.

However, it should be emphasised that the 'semantics' of these category labels (in the sense of the set of objects in corpus data they refer to) is variable and above all language-dependent. There can be no assumption, for example, that the Noun Phrase as used in one annotation scheme will map straightforwardly on to the Noun Phrase as used in another. (See further the discussion of 'parsing schemes"in section 6.)

In addition to these examples, and by way of further illustration, in Appendix A we will present short passages (of about ten sentences) from some of the EU languages covered by this document. It is recommended that groups that produce an annotated corpus minimally document the corpus in a similar way (see section 6). That is, they should list the categories used and provide an example of each; they should also provide at least a small stretch of text illustrating the application of the annotation scheme.

The recommendations that we make are expressed in a form which applies to phrase structure models of annotation. Examples of dependency-based models are few, and it would be premature to formulate recommendations for dependency-based annotation. However, reference should be made to the Helsinki Constraint Grammar as a key exemplar.

If a phrase structure annotation is adopted, we recommend the following categories:

- Sentence
- Clause (with qualifications — see sections 4.2.1 and 4.2.2 below)
- Noun Phrase
- Verb Phrase
- Adjective Phrase
- Adverb Phrase
- Prepositional Phrase

In the following subsections we will define and illustrate these labels and structures. We will also mention some tests that can be used to determine a number of constituents. (Such tests can be found in a number of introductory textbooks, e.g. Radford 1988.)

It should be noted that the examples illustrating these labels are meant to illustrate only the labels themselves and not to provide a complete annotation.

4.2.1 Sentence (S)

The maximal, syntactically independent, segments into which a text is subdivided, for parsing purposes, are normally considered to be sentences. In a written text, they are typically (though by no means invariably) delimited by an initial capital letter[10] and a final full stop ('.') or other terminal punctuation. It is convenient to accept this primary orthographic definition of 'sentence' for the purposes of syntactic annotation. However, a sentence, so defined, may be either a full sentence:

(9) [S This is a sentence. S]

or a 'grammatically incomplete' one:

(10) [S Well done. S]

The same applies to sentences included within other sentences, as in:

(11) [S [S "Well done", S] she said. S]

"Well done" in (11) is labelled as a sentence, since it clearly has an independent syntactic status equivalent to those of (9) and (10), even though it is included in another sentence. This inclusion of one independent sentence within another is found both with reported speech and elsewhere. Phenomena such as those illustrated in (10) and (11) are by no means exceptional in text corpora.

In transcriptions of spoken discourse, there is no simple answer to the question "What is a sentence?". Some transcriptions, based on standard orthography, yield *de facto* sentences in the form of units beginning with a capital letter and closing with a terminal punctuation mark. For these, there is no problem in recognising the primary sentential segments and delimiting them by [S . . . S], even though these segments frequently lack the canonical structure of a complete written sentence. Moreover, even in other transcriptions, where the standard orthographic practices of sentence delimitation are avoided, it is possible to identify 'primary segments' analogous to the written sentence, *viz.* the primary units into which the transcribed discourse is divided for parsing purposes. For spoken as well as written language, then, the [S] unit may be retained, although it may be interpreted differently, and some other term, such as 'primary segment', may be preferred to 'sentence'.

We conclude by recommending, for the syntactic annotation of any text (including a transcription of spoken language), an exhaustive division of the text into units labelled [S . . . S][11].

4.2.2 Clause (CL)

Traditionally sentences are deemed to be composed of major constituents known as *clauses*, which may be main clauses (not included in a superordinate clause) or embedded clauses (included in a superordinate clause) such as relative clauses or adverbial clauses. A main clause, unless it is coordinated, is equivalent to a simple sentence, and does not need to be separately labelled. Embedded and coordinated clauses, on the other hand, will need to be separately identified. We recommend that such units be identified in the annotation, and labelled either as sentences (S) or as clauses (CL), according to the preference specified in the annotation scheme.

The reason for allowing choice here is that different theoretical preferences have to be accommodated. In some syntactic models, the 'clause' category is not used (except informally), embedded clauses being marked by included [S] constituents. In other models, clauses are identified as such, even where they are coextensive with an independent sentence.

One solution which commends itself (and is employed in the Lancaster Treebank and the SUSANNE Corpus) is to retain [S . . . S] as the delimiter of sentences, whether included or not, as defined in section 4.2.1 above, and also to use [S . . . S] for the coordinated parts of a compound sentence; but to use 'clause' labels for subordinate clauses. An example which illustrates this division is (12)[12]:

(12) [S [S The distinction at issue is relatively clear S] , but [S closer examination reveals [CL that all is not quite so straightforward [CL as it seems CL] CL] S] . S]

Embedded (subordinate) clauses are generally identified by some sign of their dependence on the superordinate clause or sentence, such as an introductory conjunction or a non-finite verb form. It is noticeable that the coordinated parts of the compound sentence do not have to be clause-like units with subjects and verbs, and to this extent [CL] would be a misleading label to apply to them. This is true, in (13), of *so far so good* in:

[10] Of course,the initial capital letter may in some cases be preceded by sentence initial punctuation marks, such as the initial inverted question mark preceding Spanish questions.

[11] By 'exhaustive' we mean that all tokens which are part of the verbal content of the text should be included in an [S . . . S]. This recommendation does not apply, of course, to words or symbols which are part of the mark-up of the text, such as those representing (in spoken transcriptions) pauses or the beginning and ending of a speaker's turn.

[12] Throughout these guidelines, as in this example, the syntactic annotations are purely illustrative, and are not meant to serve as a model to be imitated in all details.

(13) [S [S So far so good S] , but [S now consider gender in adjectives S] . S]

4.2.3 Noun phrase (NP)

At phrase level, the noun phrase is probably the least problematic of the categories to be dealt with. In general, a noun phrase will a have noun or a pronoun as its head, and included within the noun phrase are the determinative elements, any premodification, and any postmodification. The examples below, (14) to (17) show noun phrases with the head noun/pronoun in bold:

(14) [NP **He** NP] was a tiny man

(15) [NP his white **shirt cuffs** NP]

(16) [NP his surprisingly thick and hairy **wrists** NP]

(17) [NP some wholly unanticipated but remotely possible **event** of absorbing interest NP]

However, noun phrases may also occur with adjectival heads, as in (18) and (19):

(18) [NP The **unemployed** NP] have had enough

(19) We've beaten [NP the **best** NP]

or with a head which is a cardinal or ordinal number, as in (20) and (21:

(20) [NP The **ninth** NP] is my particular favourite

(21) [NP The other **seven** NP] continued with the trip

In 'pro-drop' languages, such as Spanish and Italian, pronominal Subjects are usually not expressed. Depending on the chosen type of analysis, this may require another definition of noun phrase, in order to include 'empty noun phrases', in which the pronoun is not actually present, but may be inferred from the verb ending.
A classic constituency test for Noun Phrases is that only whole NPs can be moved within the same sentence. In English, constituents can be preposed to achieve some effect, as in (23) (from Radford 1988: 70):

(22) I can't stand your elder sister

(23) *Your elder sister* I can't stand (though your brother's OK).

Examples (24) and (25) show that it is not possible to move only part of the NP:

(24) *Your elder I can't stand sister

(25) *Elder sister, I can't stand your

However, this test should be used with caution. It works well in English, but not always in other languages. For example, in (26) *Neue Bücher* is moved to the beginning of the sentence while *keine* is left at the end:

(26) Neue Bücher habe ich keine
new books have I no
'I have not got any new books'

4.2.4 Verb phrase (VP)

This category is slightly more difficult to define, since there is disagreement over the extent of the verb phrase. In particular, should the verb phrase include only the words that are verbs, or should it also include the complements of the verb? In the examples given in this document, and in the sample texts in the appendices, we have chosen to include the complements, but it must be noted that this is an open issue, and we are in no way implying that this analysis is preferable to the alternative. The choice to be made at this level, i.e. the inclusion or exclusion of verbal complements in the Verb Phrase, is shown by the examples in (27) and (28), (27) showing the inclusion of the complement of the verb in the verb phrase and (28) excluding the complement:

(27) He [VP took up [NP a clothes brush NP] VP]

(28) He [VP took up VP] [NP a clothes brush NP]

An advantage in the type of analysis shown in (27) is that the relative levels of the constituents can be shown to a greater extent — i.e. complements of the verb are included in the verb phrase, while adjuncts and peripheral adverbials are left at sentence level.
However, in a case where an adjunct occurs before the complement of the verb, the approaches used in (27) and (28) would cause problems, since either both the adjunct and the complement would be included as daughters of the verb phrase, or both would be daughters of the sentence, rather than keeping the complement as a daughter of the verb phrase and the adjunct as a sister of the verb phrase. These problems may be solved by an additional notation, but at some level, arbitrariness is inevitable.
Regardless of the choice made over the extent of the Verb Phrase, there arises a problem of discontinuous Verb Phrases. A complex verbal construction may be discontinuous, e.g. the auxiliary and the main verb are separated in inverted constructions in English, or the main verb is positioned at the end of the sentence in German and Dutch. Such discontinuity can be avoided by having different labels and constituents for the auxiliary verb and the main verb, resulting in an analysis as shown in the Dutch example (29) below:

(29) [NP Ze NP] [AUX zullen AUX] [ADVP er ADVP] [VP [NP de VN-agenda [PP voor [NP het komende jaar NP] PP] NP] behandelen VP].

and in the English interrogative inverted example (30), using the so-called ‘dummy auxiliary’ *do*:

(30) [AUX Do AUX] [NP they NP] [VP confide [PP in you PP] VP]?

As with Noun Phrases, Verb Phrases can be identified by a constituency test. In strong constituency languages like English, the whole VP can be moved, but not part of it: compare (31) and (32):

(31) Give in to blackmail, I never will

(32) *Give in, I never will to blackmail

However, there are languages in which constituent tests do not work. These will typically be languages with flexible word order, such as Finnish. (33) is an example of a discontinuous VP (Vilkuna 1989: 26):

(33) Maailmaa nähnyt hän on.
world-Part seen he is
‘He IS a widely-travelled person.’

For Finnish, then, evidence for a VP is less convincing than it is for English, and a dependency approach seems the more natural choice. (Covington (1990) provides a parsing strategy for variable word order languages and Covington (1991) for parsing discontinuous constituents, both using a dependency syntax approach.) See Section 4.3.4 for further discussion of discontinuous constituents.
In Italian also, constituency tests cannot be applied. This can be shown through the distribution of VP-adverbs (e.g. *completamente* ‘completely’, *intenzionalmente* ‘intentionally’, *attentamente* ‘carefully’) and S-adverbs (e.g. *probabilmente* ‘probably’, *certamente* ‘certainly’). In English, these different classes of adverbs have a different distribution within the sentence. In contrast, in Italian, the distinct adverb classes cannot be distinguished on the basis of their distribution in the sentence. S-adverbs and VP-adverbs can occur in the same positions within the sentence, as illustrated in examples (34) to (37):

(34) *Attentamente/certamente, il bambino ascoltó la storia*
‘Carefully/certainly, the child listened to the story’

(35) *Il bambino attentamente/certamente ascoltó la storia*
‘The child carefully/certainly listened to the story’

(36) *Il bambino ascoltó attentamente/certamente la storia*
‘The child listened carefully/certainly to the story’

(37) *Il bambino ascoltó la storia attentamente/certamente*
‘The child listened to the story carefully/certainly’

Thus, in Italian as well as other languages, neither the position nor the syntactic context can help to decide whether an adverb is an S-adverb or a VP-adverb; this can only be stated by considering its semantic content and the way it relates to the content of the predicate or the sentence. This situation has consequences for the success of standard VP-tests.

4.2.5 Prepositional phrase (PP)

A sequence of a preposition and its complement is a prepositional phrase. The complement of a preposition is usually a noun phrase (see examples (38) to (40)), but may also be a clause or an adverb phrase. According to the categories recommended here, a prepositional phrase may be analysed further into preposition and noun phrase. The examples below demonstrate how this further analysis can be a recursive procedure.

(38) [PP en [NP sustitucion [PP de [NP los canales correspondientes [PP de [NP 50 baudios NP] PP] NP] PP] NP] PP].

(39) [NP Fairbanks NP] [VP hummed [NP a few bars NP] VP] [PP in [NP a voice [VP made resonant [PP by [NP the very weakness [PP of [NP his chest NP] PP] NP] PP] VP] NP] PP].

(40) [PP En [NP el caso [PP de [NP un sistema mixto [PP en [NP el [CL que [VP se utilicen [NP canales [PP con [NP tres velocidades [PP de [NP modulacion NP] PP] diferentes NP] PP] NP] VP] CL] NP] PP] NP] PP] NP] PP]

In a language such as Spanish, where a large proportion of the modification of nouns takes the form of a following preposition *de* and another noun, this recursion is extremely prevalent, as in (40). In cases where the prepositional phrase is complemented by a one word noun phrase, it may be advantageous to leave the analysis at this point, rather than continuing to analyse further by enclosing the complement (see also section 5.4 on one-word constituents).

4.2.6 Adverb phrase (ADVP)

An adverb phrase may consist of an adverb, or a sequence of words in which an adverb is the head of the phrase. Adverb phrases may function as adverbials, as in (41):

(41) [NP Her beautiful white hat NP] [VP was [ADVP very nearly ADVP] ruined VP]

or as modifiers of adjectives, as in (42):

(42) [NP Il NP] [VP parle [ADVP infiniment plus couramment ADVP] VP]

or noun phrases, as in (43):

(43) [NP They NP] [VP let [NP me NP] [VP speak VP] [ADVP now and then ADVP] VP]

or as the complement of a preposition, as in (44):

(44) [ADVP Strangely enough ADVP] , [NP we NP] [VP received [NP a reply NP] [NP the next day NP] VP]

Other examples:

(45) [NP The book NP] [VP is [ADVP right here ADVP] VP]

(46) [ADVP Como [NP resultado [PP de [NP esa trama NP] PP] NP] ADVP] [VP no se lleva [PP a cabo PP] [NP ninguna acción NP] VP]

4.2.7 Adjective phrase (ADJP)

An adjective phrase may consist of an adjective, or a sequence of words in which an adjective is the head of the phrase, as shown in (47) to (50) below.

(47) [NP his [ADJP surprisingly thick and hairy ADJP] wrists NP]

(48) [NP some [ADJP [ADJP wholly unanticipated ADJP] but [ADJP remotely possible ADJP] ADJP] event NP]

(49) [S [NP His speeches NP] [VP are [ADVP always ADVP] [ADJP too long [PP for comfort PP] ADJP] VP] S]

(50) [AUX have AUX] [NP you NP] [VP found [NP something [ADJP suitable [PP for [NP your needs NP] PP] ADJP] NP] VP] ?

4.2.8 Coordination

As has already been shown in some of the preceding examples, the issue of coordination necessarily arises: how is coordination to be represented in terms of constituency? Different approaches have been taken, and in the example analyses given in this document, we have chosen to take a traditional approach, showing the coordinated constituents at the same level, with the conjunction between them (see also (47) and (48)):

(51) [NP [NP John NP] and [NP Mary NP] NP]

(52) She went [PP [PP to the library PP] or [PP to the cafeteria PP] PP]

(53) He works [ADVP [ADVP very slowly ADVP] but [ADVP very meticulously ADVP] ADVP]

However, in practice, in an automated parsing system, this is not an easy differentiation to make, and in some existing schemes, a slightly less satisfactory solution has been found, *viz.* analysing coordination in a similar fashion to subordination.

Most constituents (both phrases and clauses) can be coordinated, but the extent to which this is possible will differ across languages. The conjuncts may be marked as such by separate descriptors: NP_{co} etc. However, there are many occasions where the conjuncts are not of the same formal category, or where they do not correspond to an entire phrasal or clausal constituent. There is much to be said, in these cases, or perhaps for all cases of coordination, for the use of a generalised label applied to all coordinate constituents or conjuncts, e.g. the label CO used in the TOSCA system. We do not offer a definitive solution for the annotation of coordination, and the many variants of coordination will not be considered further in this report. See Sampson (1995: 310f) for a detailed treatment.

4.3 Optional annotations

This section is divided into three subsections:

1. optional syntactic annotations
2. optional semantic annotations
3. borderline categories, in which features could be described as either/both syntactic and semantic (see section 2.2.5 on the syntax/semantics boundary)

4.3.1 Optional syntactic annotation

An annotation scheme which confined itself to the recommended category labels in section (4.2) above would give a sparse and incomplete representation of the syntactic form of sentences in a corpus. For many purposes and for many languages, other major constituent labels, such as Auxiliary, Determiner Phrase, or (in English) Genitive Phrase will be found to be necessary. We will say no more of these here.

4.3.1.1 Sentence subcategorisation

It is common practice in most existing schemes to further subcategorise sentences. [S] can be used for all sentences, or may be restricted to simple declarative sentences or imperatives, and questions with declarative word order. Other labels may be introduced, e.g. [SI] may be used to mark sentences in which the verb is imperative; [SQ] to mark questions, including such constructions as ‘yes/no’ questions, or ‘tag questions’. Other descriptors may be required in order to mark sentences of kinds which are included in other sentences, e.g. ‘direct quotations’ or ‘interpolated sentences’. Whenever the need for such refinements arises, they can be introduced into the annotation scheme. These additions should of course be documented.

4.3.1.2 Syntactic clause subcategorisation

In section 4.2 above `Clause` was introduced as a recommended category . Clauses may be further subclassified traditionally by syntactic properties (such as whether the verb is finite or non-finite, or the clause verbless), or by functional properties (e.g. whether an embedded clause is `nominal`, `adverbial`, `relative` or `comparative`; see on borderline categoriessection 4.3.3). Further possibilities for subcategorisation may be language-specific. For example, in English it is common practice to categorise clauses according to their introductory subordinator, e.g. *that*-clauses, or *wh*-clauses.

At this level of analysis, the most useful distinctions are the purely syntactic ones of `finite`, `non-finite`, and `verbless`. The functional properties mentioned above are more dependant on semantic roles of the clause in question, and will therefore be dealt with later. These syntactic distinctions may be further subdivided: e.g. under `non-finite` one can distinguish between `infinitive clauses`, `gerundival` or `participial clauses`, and `past participial clauses`.

4.3.1.3 Syntactic phrase subcategorisation

Phrases may be further subcategorised to show syntactic features such as `gender`, `person` and `number`. Thus a Noun Phrase may be marked as singular, masculine, etc. However, as morpho-syntactic tagging currently tends to be a preliminary to syntactic annotation, this type of information can often be derived from the POS-tag of the head of the phrase. The grammatical features of the head of the constituent may therefore be percolated up to the highest node of that constituent.
In (54) and (55), the subcategorisation features of the Noun *filles* in (54) — 3rd Person Plural Feminine:

(54) [NP Les filles_N3PlFem NP] [VP ont écrit [NP les lettres NP] VP]

have been transferred from word to phrase level in (55):

(55) [NP Les filles NP-3PlFem] [VP ont écrit [NP les lettres NP] VP]

In coordinated NPs one or more of these features may differ. Languages vary as to which feature dominates the phrase. In French, for example (see (56) and (57)), when one conjunct in a coordinated NP is masculine singular and another feminine singular, the phrase as a whole will be marked as masculine plural:

(56) [NP[NP Le garçon_N3MascSg NP] et [NP la fille_N3FemSg NP] NP] . . .

(57) [NP[NP Le garçon_N3MascSg NP] et [NP la fille_N3FemSg NP] NP-3MascPl] . . .

Syntactic subcategorisation of other phrases may also occur. For example, a Verb Phrase may be marked according to tense, mood, aspect, voice, etc., as well as according to the complementation frame (transitive, copular, intransitive etc) it represents.

4.3.1.4 Grammatical function

Optionally, syntactic functions can be assigned to constituents. At the rank of sentence or clause, the Lexicon/Syntax SubGroup mark only one grammatical function, namely +/–Subject, but other grammatical functions may be derived from the combination of several other syntactic features plus the property +/– Subject[13]. For annotated corpora, we propose Subject, Object, Indirect Object (if it applies in a language) and Adjunct. If used, these labels may be hyphenated to the constituent label, as in (58):

(58) [S [NP-Subj John NP-Subj] [VP gave [NP-Obj a book NP-Obj] [PP-IndObj to [NP Mary NP] PP-IndObj] [PP-Adjct on [NP Monday NP] PP-Adjct] VP] . S]

At the phrase level, other functions, such as head and modifier are commonly recognised.

4.3.2 Semantic annotation

4.3.2.1 Semantic phrase subcategorisation

Phrases may be subcategorised according to semantic features; for example, a Noun Phrase may be marked as `definite`, or `indefinite`, or as `locative` or `temporal`. This type of attribute corresponds directly to the syntactic subcategorisationmentioned in section 4.3.1. If the corpus is previously morphosyntactically annotated, this information is often included at the level of POS tagging.
Other semantic subcategorisation of phrases may occur when a phrase (or clause) has been identified as an `adverbial` (see section 2.2.5). Adverbials may be marked according to their semantic function, e.g. `time`, `manner`, `place`, `purpose`, `reason`, `condition`, `comparison`, `modality`, etc.

4.3.2.2 Semantic clause subcategorisation

Similarly, clauses that are identified as `adverbials` may be subcategorised semantically, as adverbial clauses of time, of condition, of reason, etc. Other subcategorisation of clauses is mainly functional, and this will be dealt with in section 4.3.3.

[13] At this time, the representation of the Lexicon/Syntax Subgroup is still in progress, but the properties of syntactic features such as case and +/– predicative combine together to give grammatical function information such as Object (from the combination of features: −-subject, –predicative, case=accusative), Subject Predicative Complement (from the combination of features: +subject, +predicative) etc.

4.3.3 Syntactic/semantic (functional) annotation

4.3.3.1 Clause functional annotation

Embedded clauses may be further subclassified according to their function within the superordinate clause. Here, syntactic and semantic criteria may overlap or coincide. Possible subdivisions at this level are the following:

4.3.3.1.1 Nominal clause

Nominal clauses can function in some or all of the noun phrase functions, such as Subject, Object, Complement, etc. These include *that*-clauses, interrogative clauses and gerundive (*-ing*) clauses, as in (59):

(59) [S [NP He NP] [VP enjoys [CL-NOM playing practical jokes CL-NOM] VP] . S]

4.3.3.1.2 Adverbial clause

Adverbial clauses function mainly as adjuncts. Semantically, they distinguish approximately the functions that adverbs show, as in (60):

(60) [S [NP We NP] [VP left [CL-ADV after the speech ended CL-ADV] VP] . S]

4.3.3.1.3 Comparative clause

Comparative clauses resemble adjectives and adverbs in their modifying functions, as in (61):

(61) [S [NP He NP] [VP is as clever [CL-COMP as I thought CL-COMP] VP] . S]

4.3.3.1.4 Relative clause

A relative clause generally functions as a modifier of a noun, as in (62):

(62) [S [NP The book [CL-REL that I bought yesterday CL-REL] NP]. S]

4.3.4 Annotation of deep/logical information

Under this heading fall a number of phenomena that can be collectively referred to as discontinuity and trace phenomena, i.e. movement phenomena where an element has been moved (or removed) from its 'logical' position. In (63) is an example in which a relative clause has been moved from its logical position in an NP:

(63) A man was at the door who wanted to speak to you

Here, the relative clause *who wanted to speak to you* is moved to the end of the sentence. An index system may be used to indicate the position that the relative clause 'logically' occupies. This is shown in (64) (where a number of details have been left out for the sake of clarity):

(64) [NP A man [*1] NP] [VP was [at the door] VP] [CL-Rel*1 who wanted to speak to you CL-Rel]

Indices may also be used for zero constituents, such as the zero relative pronoun in the case of some relative clauses. There may also be a need to include such traces in syntactic annotation even where no discontinuity occurs. A classic example is found in English when a preposition occurs in a final position, without a following prepositional complement. In (65), the trace constituent [*1] is indexed to show the discontinuity of the prepositional phrase:

(65) [S [NP Who NP*1] [Aux does Aux] [NP she NP] [VP work [PP for [*1] PP] VP] ? S]

In (66), a superficially similar construction, there is no discontinuity, but the missing prepositional complement is identified by the unindexed trace constituent [*0]:

(66) [S [NP He NP] [VP has [NP nothing [VP to live [PP for [*0] PP] VP] NP] VP] . S]

This zero constituent may also be used to indicate ellipsis, where no movement phenomenon occurs.
Indices can also be used to indicate 'underlying' Subject or Object postions in the complement of so-called raising constructions and with control verbs. In (67) *James* is the Subject of *tried* but logically, it is also the Subject of *to go*:

(67) [NP James NP*1] [VP tried [[CL [*1] to go CL] VP]

In (68) *John* is the Object of *believed* and logically, also the Subject of *be*:

(68) [NP Mary NP] [VP believed [NP John NP*1] [CL [*1] to be at home CL] VP]

Using indexes as shown here, these logical relations may be made explicit.
Languages differ in the range of possibilities of discontinuity, raising constructions and control phenomena (e.g. in Raising-to-Object constructions, the Dutch equivalent of *believe* can only take tensed complements, not infinitival ones as in English). However, each language will probably have a number of constructions in which logical relations need to be indicated using an index system. It is beyond the scope of this report to go into further detail, but the ideas set forth here should enable a simple indexing system to be implemented. Details of two actual applications of such systems can be found in the documentation of the SUSANNE corpus (Sampson 1995, Ch. 5) and the Penn Treebank (Marcus et al. 1994).

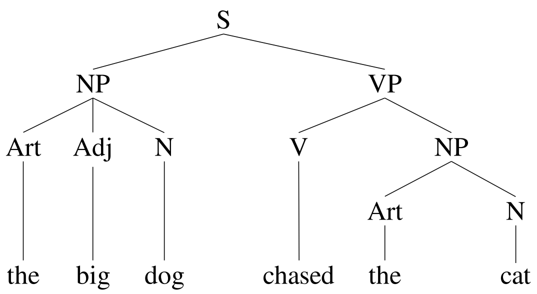
5 Issues in practical application

Once a basic scheme for annotation has been arrived at, many issues will arise in the application of the scheme. In this section, some of the more important issues are discussed, and in most cases no actual recommendations are made. In the formation of a more detailed annotation scheme these types of issues will arise frequently, and the choices to be made are to a large extent unimportant, as long as a choice is made and is adhered to consistently. What *is* important is that such decisions must be documented in detail (see section 6).
The issues discussed in this section are the following:

- Phrase Structure or Dependency Representation (5.1)
- Underspecification, Ambiguity and Ambivalence (5.2)
- Dealing with punctuation (5.3)
- Single-word constituents (5.4)

5.1 Phrase structure *vs* dependency

There are two major varieties of syntactic annotation: a phrase structure and a dependency representation. In general, a phrase structure representation may be found more suitable for languages with rather fixed word order patterns and clear constituency structures. Dependency representations, in contrast, may be found more adequate for languages which allow greater freedom of word order and in which linearisation is controlled more by pragmatic than by syntactic factors. This is the case in Finnish (and the Slavonic languages, some of which, such as Czech and Polish, may be expected to have increasing association with the EU in the future). Less obviously, some Romance languages (Italian and Spanish) may also benefit from a dependency representation. However, this does not mean that languages such as English *should* be annotated using a phrase structure representation or, *vice versa*, that for languages with greater freedom of word order, dependency *should* be used. Indeed, dependency structures have been successfully applied to English using the English Constraint Grammar (Karlsson *et. al.* 1995) and the Slot Grammar Parser (McCord 1990).
Since the approach to syntactic annotation is to a large extent influenced by the language to be annotated, our guidelines do not give any preference either to a phrase structure annotation or to a dependency annotation. The phrase structure annotation, however, is in certain ways the more demanding of the two, which is why this report covers phrase structure in more detail. This should not be construed, however, as expressing a preference for phrase structure annotation.
The two possibilities mentioned here, Dependency and simple Phrase Structure grammar models, are certainly not the only options available to annotate a corpus. Other approaches, such as LFG and complex phrase structure grammar models such as GPSG and HPS, may be equally successful. However, the reason why only phrase structure and dependency grammars are covered here is that by now these two models have a certain tradition in corpus annotation; and they have been used to annotate corpora both manually and automatically. Though it is true that HPSG parsers exist, there are no corpora, as far as we know, annotated using a HPSG formalism, nor are there any existing HPSG parsers robust enough and of sufficiently wide coverage to serve as a basis for corpus annotation.
We will propose notations for both approaches. A typical Phrase Structure tree is shown in (69):

(69)


The constituent structure and the labels can also be represented in a labelled bracketed structure, as in (70):

(70) [NP The big dog NP] [VP chased [NP the cat NP] VP]

We propose that labels be put both at the opening and at the closing brackets. Though this may lead to many labels per constituent, it is nevertheless preferable for purposes of readability in order to distinguish multitudes of brackets. Compare (71), (72) and (73).

(71) [PP in [NP the heat [PP of [NP the night]]]]

(72) [in [the heat [of [the night NP] PP] PP] NP]

(73) [PP in [NP the heat [PP of [NP the night NP] PP] NP] PP]

The labelling as in (71) is used in the Penn Treebank and partially in TOSCA: the contituents are labelled only at the opening brackets. The scheme in (72) is not used, as far as we know. The one in (73) is used in SUSANNE and in the IBM Paris scheme. We propose the use of the double-labelling convention of (73) for the sake of clarity.
Dependency trees can be represented with arrows pointing from the head to the dependents or from the dependents to the heads (see also section 3.3 for details). Of these two conventions, we recommend the use of the latter, as in (74):

(74) The big dog chased the cat

Arrows 'depart' from dependents and point to their head. Thus, in (74) *The* and *big* are dependents of *dog*. We propose a scheme in which the words of the sentence are in a column, preceded by a column with word numbers. The dependencies can be indicated using these numbers as a reference. Further columns may be added for syntactic labels. Using this scheme, (74) is represented as in (75)[14]:

(75)	1	The		3
	2	big		3
	3	dog	Subj	4
	4	chased	MainV	
	5	the		6
	6	cat	Obj	4

In this scheme, there is actually no need for arrows since the dependencies and their direction can be inferred from the numbers in the fourth column: if the number in the fourth column is bigger than the number in the first column, the word's head follows, otherwise it precedes. For example, *The* is word number 1 with dependency index 3, so its head (*dog*, the third word) follows. Similarly, *cat* is word 6 with dependency index 4, so its head precedes. Note that the head of the sentence (*chased*) has no entry in the fourth column.
We note further that an alternative dependency representation has been suggested. Since dependency trees are directed a-cyclic graphs they can be represented by bracketed expressions, just as constituency trees. The governing term is placed first, and is enclosed in brackets, including all dependent terms which are themselves included in brackets. Example (74) can then be represented as in (76) (with part of speech categories added):

(76) [V chased [N dog [det the] [Adj big]] [N cat [Det the]]]

This bracketed string can be represented more perspicuously as in (77) (with functional labels added; note that the indented format implies some form of constituency):

```
(77) [PRED chased
         [SUBJ dog
             [DET the]
             [ATTR big] ]
         [DIROBJ cat
             [DET the] ] ]
```

This method can also be used for annotating the running text, as in (78), with the word tokens in their order of occurrence:

(78) [PRED [SUBJ [DET the] [ATTR big] dog] chased [DIROBJ [DET the] cat]]

However, a disadvantage of this latter method is that it does not work for discontinuous structures, and since representing discontinuous structures is the strength of the vertical representation, it is the preferred one for languages in which discontinuity is a recurrent phenomenon.
As with phrase structure representations, less fine-grained analyses are possible too, as in (79) (see section 5.4 for discussion of bracketing single word constituents):

(79) [PRED [SUBJ the big dog] chased [DIROBJ the cat]]

5.2 Underspecification, ambiguity and ambivalence

Underspecification, ambiguity and ambivalence are similar phenomena, in that they both amount to an 'information gap' in the annotation of a corpus. From the present point of view, underspecification is the easiest phenomenon to deal with, and will be discussed first.

[14] Other columns can be added to hold other types of information, such as POS tags.

5.2.1 Underspecification

This is a phenomenon which can be characterised as incompleteness of annotation. That is, a piece of information that could have been specified in the annotation of a corpus is simply lacking, for one of a number of possible reasons.
Underspecification, at one level, is a necessary and even benign phenomenon. Not all syntactic details (e.g. of subcategorisation of Noun Phrases or Verb Phrases) can be included in any realistic corpus annotation task. It is normal for a parsing scheme to be selective in specifying the syntactic phenomena to be identified and labelled, and omission of some details is almost inevitable.
At another level, underspecification may show unevenness in the degree of detail achieved, and this may be an unavoidable consequence of limitations of human resources, time, software capabilities, etc. For example, it may be possible to mark pronominal Noun Phrases for number, but impossible to mark full Noun Phrases for number, within an acceptable threshold of accuracy. In this case, number will be marked only for some Noun Phrases. Another case of a similar kind occurs where some of the constituents of a sentence are labelled, and others are left unlabelled. This kind of 'information gap' should be documented in the annotation scheme, and inconsistency in its occurrence should if possible be avoided.

5.2.2 Ambiguity

This phenomenon we define as an unresolved choice between alternative annotations. Again, there may be a number of reasons for ambiguous syntactic annotation[15], some motivated by practical considerations, and others by more theoretical linguistic considerations. Here we distinguish two major kinds of ambiguity.

5.2.2.1 System ambiguity

A system ambiguity is an ambiguity which results from parser limitations. Automatic corpus parsers, which are likely to be increasingly employed for syntactic annotation on a large scale in the future, are at present far from achieving the 'ideal' outcome of a single correct parse for each sentence. Where there is likelihood of parsing error, a preferable solution, for most purposes, will be for the parser to output more than one parse, leaving it to the human interpreter to decide which is a correct analysis. Thus, the Helsinki ENGCG parser leaves alternative analyses in the output, where disambiguation cannot be achieved. This is illustrated in table 10, where, due to the categorical ambiguity of *addresses*, two syntactic structures are possible: [NP this paper NP] [VP addresses ...] and [NP this [paper addresses] NP].

Table 10: Indicating ambiguity in ENGCG

```
”<*this>”
      ”this” <*> DET CENTRAL DEM SG @DN>
”<paper>”
      ”paper” N NOM SG @SUBJ @NN>
”<addresses>”
      ”address” <SVO> V PRES SG3 VFIN @+FMAINV
      ”address” N NOM PL @NPHR
”<three>”
      ”three” NUM CARD @QN>
”<main>”
      ”main” A ABS @AN>
”<issues>”
      ”issue” N NOM PL @NPHR @OBJ
```

Although in principle a manual posteditor could eliminate most system ambiguities, in practice such a postediting phase may be too expensive or too time-consuming, so that ambiguities will have to remain in the annotated corpus.

5.2.2.2 Use ambiguity

This second kind of ambiguity is the kind that cannot be resolved by a human posteditor — because even when context is taken into account, the sentence cannot be disambiguated by a human interpreter. Such 'genuine ambiguities' may result in a glaring difference of meaning, as in (80) and (81), where it is not clear whether Stephen Glanville is Agatha Christie's husband or not:

[15] Since we are here concerned with syntactic ambiguity only, we ignore types of ambiguity (e.g. purely lexical ambiguity or pragmatic ambiguity) which have no bearing on syntactic annotation.

(80) Stephen Glanville, who died in 1956, was [NP an Egyptologist friend [PP of [NP [NP Agatha Christie NP] and [NP her husband NP] NP] PP] NP]

(81) Stephen Glanville, who died in 1956, was [NP [NP an Egyptologist friend [PP of [NP Agatha Christie NP] PP] NP] and [NP her husband NP] NP]

On the other hand, a surprising number of use ambiguities make relatively little difference to meaning, as in (82) and (83):

(82) They [VP let me [VP speak [ADVP now and then ADVP] VP] VP]

(83) They [VP let me [VP speak VP] [ADVP now and then ADVP] VP]

In (82) and (83), the adverbial *now and then* can equally well modify *let me [speak]* or just *speak*[16].
Our recommendation is, where feasible, to indicate ambiguities in the annotated corpus, and to specify by an annotation device whether the ambiguity is of the system type or the use type. The nature and function of the annotation device should be specified in the documentation to the annotation scheme.

5.2.2.3 Methods of indicating ambiguities

Syntactic ambiguities may be of various types: in fact, all the layers of syntactic annotation listed in section 3 (bracketing of segments, labelling of segments, subcategorisation of segments, etc.) may give rise to ambiguity. Most cases of ambiguity, however, can be reduced to (a) ambiguities in the configurations of nodes in a syntactic tree, (b) ambiguities in the labelling of nodes, and (c) ambiguities combining (a) and (b). In a phrase structure model of annotation, these can be simply listed as (a´) bracketing, (b´) labelling and (c´) bracketing + labelling[17].
As an indication of how ambiguities may in practice be included in an annotation scheme, we illustrate one method of specifying alternative analysis in a linear (horizontal) format for phrase structure annotation. (The actual choice of an ambiguity-encoding device is a matter for the Text Representation guidelines.) The device we illustrate can be symbolised

// x1 / x2 / . . . / xn //

where // delimits the whole ambiguity set, / separates one alternative analysis from another, and x1, x2, . . . xn represents the alternative analyses. The same device can be used for alternative labellings, alternative bracketings, and alternative labellings + bracketings. The method of representation least open to misunderstanding is the one where x1, x2, . . . xn are entire sentences. But this method also suffers from prolixity, and probably a more satisfactory method can be arrived at by 'factoring out' from the whole sentence the parts which give rise to ambiguity and the parts which have a constant interpretation. This will clearly be the most practical solution where the ambiguity is purely a matter of labelling. As an example of the proposed notation, in (84) the readings of both (80) and (81) are indicated:

(84) Stephen Glanville, who died in 1956, was // [NP an Egyptologist friend [PP of [NP [NP Agatha Christie NP] and [NP her husband NP] NP] PP] NP] / [NP [NP an Egyptologist friend [PP of [NP Agatha Christie NP] PP] NP] and [NP her husband NP] NP] //

It should be noted, finally, that ambiguity and underspecification are technically different (the one spelling out alternatives, and the other leaving alternatives implicit), but that, in practice, they may amount to the same thing. For example, in dependency tree annotation (see section 3.3) it may be found convenient (as in current versions of ENGCG) to omit identification of the head of a dependency link, while specifying its word class. This omission fits the definition of underspecification, but it would be possible to identify all possible heads in the sentence for a given dependent, and hence to state explicitly the alternative analyses, thus reformulating underspecification as ambiguity.

5.2.3 Ambivalence

To prevent confusion in a further direction, it is important to distinguish ambiguity from ambivalence. For our present purpose, ambivalence denotes a phenomenon of uncertainty as to which annotation should apply to a given sentence, not because the sentence itself is ambiguous in this respect, but because the semantics of the annotation scheme is inexplicit (see section 6 below on documentation). For example, in the sentence *She's older than her sister*, it may be unclear whether *than her sister* is to be treated as a prepositional phrase or as a comparative clause (or perhaps as some

[16] A third type of use ambiguity occurs where the human interpreter cannot make sense of the syntax of the sentence. For example, the following sentence from a computer manual cannot be readily interpreted or parsed by a non-specialist in computing: *In this situation, the operator must ready a spool volume and IPL again.*

[17] We exclude from consideration here ambiguities of part-of-speech tagging at word level. A special case of labelling ambiguity, these have been dealt with in draft guidelines for morphosyntactic annotation (see EAGLES (1996a)).

other constituent). Ideally, it may be argued, an annotation scheme should be explicit enough to solve all problems of this kind, by stipulating a single correct analysis, however minor or rare the syntactic phenomena that may occur. In practice, however, the ideal is impossible to achieve, or at least can be achieved only by a long process of narrowing down areas of uncertainty.

An alternative position is to argue that in many areas of syntax it is unrealistic to decide on a single correct analysis: such a position can only be maintained by making arbitrary, artificial decisions. More realistic is the viewpoint that the boundaries of syntactic categories are by the nature unclear, that ambivalence is an inescapable phenomenon, and that 100% consistency in applying a parsing scheme can never be achieved.

Whichever of the two positions above is adopted, the fact remains that ambivalence is an endemic phenomenon to be coped with in corpus annotation. The way to handle ambivalence is not to include all alternative analyses in the annotated corpus, but to reduce areas of ambivalence and inconsistency as far as possible, by providing an explicit annotation scheme.

5.3 Punctuation

In the phrase structure examples given in the following sections, we have included punctuation within the bracketing. This again is a choice to be made by the scheme designers, and will probably depend on the format in which the corpus will finally be presented. However, in the syntactic annotation of written texts, there is a tradition of treating punctuation marks as 'words' for the purpose of parsing. This can be advantageous for automatic parsing systems since punctuation is typically used to mark major syntactic boundaries.

As for sentence-initial and sentence-final punctuation, it seems sensible to enclose them within the parse bracketing, as in (85):

(85) ["Good!"]

as opposed to (86):

(86) "[Good]!"

As regards medial punctuation, the most generally applicable guideline is to attach punctuation to the highest available node in the parse tree, thus assigning to medial punctuation symbols (especially commas) their value as delimiters of major constituents, as in (87):

(87) [S [NP The words [PP at [NP level four NP] PP] NP] , [PP on [NP the other hand NP] PP] , [VP are [CO [ADJP relatively rare ADJP] , and [VP not often used [PP in speech PP] VP] CO] VP] . S]

However, this guideline does not always give a satisfactory analysis of correlative punctuation, such as matching commas or dashes to indicate the opening and closing of a parenthetical constituent. Thus in (88), it makes better sense to place the second comma inside the NP, rather than to make it an immediate constituent of the sentence:

(88) [S [NP the teacher , [CL-REL who arrived late CL-REL] , NP] [VP had noticed [NP nothing NP] VP] . S]

However, our purpose here is not to dictate solutions. The principle to be adhered to is simply to make explicit in the annotation scheme whatever solution for the treatment of punctuation is adopted by the annotator. (Sampson 1995: 153f provides a considered treatment of punctuation isses.)

5.4 Bracketing of single-word constituents

A further issue that requires explicit handling in the documentation is the following. What principle is applied in deciding whether to bracket non-terminal constituents consisting of one word? Two extreme policies here would be (a) to leave unbracketed all single-word constituents, and on the other hand (b) to bracket all single-word constituents where they show their phrasal status by the possibility of adding modifers, or replacing them by a multi-word phrase.

There are difficulties with both these solutions. Solution (a) is a problem, for example, where coordination occurs between a single word and a multi-word constituent, as in (89):

(89) [NP [NP John NP] and [NP his sister NP] NP]

where, according to guideline (a), *John* would be left unbracketed, and the equivalence of the two conjuncts would not be represented in the bracketing. Solution (b) might be felt unsatisfactory because it leads to a proliferation of bracketings, as in (90), where one or more modifiers precede a noun:

(90) [NP [DETP many DETP] [ADJP recent ADJP] [ADJP unexpected ADJP] arrivals NP]

The phrase brackets delimiting the first three words here are meant to indicate the potential phrasal status of these words, because of the possibility (for example) of adding *very* before *many*.
Of these two guidelines, (b) is a problem only to the human reader. When software is used to display bracketed sentences as trees (or in any other format), the proliferation of bracketings is not an issue at all. Thus, rule (b) may be considered preferable, especially for interchange and retrieval purposes, since the filtering out of extra brackets is a much simpler task than the insertion of new ones. However, if a compromise is needed between these two extremes, a useful compromise may be to bracket single-word constituents only where they represent major constituents in the sentence, e.g. as Subject or Object, or where they are in coordination with other multi-word constituents, as in (90) above. Such conditions need to be made explicit in the documentation of the annotation scheme. (For an actual application, see Sampson 1995: 172f for an account of how single word constituents are handled in the SUSANNE corpus.)

6 Documentation and user information

As has become evident in preceding sections, an essential part of any corpus annotation project is a detailed documentation of the annotation scheme employed. (For syntactic annotation, an annotation scheme is alternatively called a *parsing scheme*.) Without documentation provided by the originators, an annotated corpus is extremely difficult for other users to apply to their own research tasks. Decisions taken in the development of the annotation scheme, as well as in its application, should be well documented in order to ensure that future users will apply the annotation scheme in a manner consistent with the originators of the scheme, and which then will be consistent in the new application. The documentation should therefore include at least the following classes of information:

1. *What layers of annotation* (in terms of layers (a)–(h)in section 3 above) *have been undertaken*. Each of these layers of annotation represents a wide area of possible annotation, and therefore more detail should be included in the documentation as to what phenomena in particular are marked by the annotation scheme.

2. *What is the set of annotation devices used* (e.g. brackets, labels).

3. *What are the meanings of these devices* (e.g. Ns = singular noun phrase; etc.). In the documentation of many existing schemes, all that is presented is a list of the labels used, and a short explanation of these symbols (e.g. VP — Verb Phrase). As has been shown in section 4.2.4 on Verb Phrases, for example, this type of explanation is not sufficient to describe the use of a label in an annotated corpus. Each symbol should be described, and illustrated with one or more examples.

4. *What are the conventions for the application of the annotation devices to texts.* A parsing scheme (or 'grammatical representation'; see Voutilainen 1994) is more than points 2 and 3 above. It includes the set of guidelines or conventions whereby the annotation symbols are to be applied to text sentences, such that (ideally) two different annotators, implementing the scheme manually to the same sentence, would agree on the analysis to be applied. In this sense, a detailed annotation scheme is a guarantee of consistency. A parsing scheme may include reference to a lexicon, to a grammar or to a reference corpus of annotated sentences. In practice it is very difficult for a parsing scheme to achieve total coverage and total explicitness in a corpus. Few corpus annotation projects have achieved anything like this. However, the nearest thing is the highly detailed parsing scheme provided by Sampson (1995) for the SUSANNE Corpus. Sampson's book discusses the various decisions taken in the application and development of the SUSANNE annotation scheme, and provides examples of the cases in which such decisions must be taken. In this respect, Sampson's book is, so far, a unique achievement.

5. *What is the measurable quality of the annotation*. This includes:

   - to what extent the corpus has been checked
   - accuracy rate
   - consistency rate

   These different measures of quality of annotation will depend mainly on how the corpus is annotated. An automatic annotation will require figures of accuracy — usually given in terms of *recall* and/or *precision*. A recall of less than 100% indicates that some appropriate readings have been discarded, while a precision of less than 100% indicates that superfluous readings remain in the output in the form of system ambiguities (see section 5.2; see Voutilainen et al. 1992 for discussion).

   A consistency rate should be given for a manual or manually post-edited annotation. Different annotators can be given a certain percentage of overlapping material, and these can then be compared to produce consistency figures. The method of comparison should also be documented.

   The extent to which the corpus has been checked overlaps to some extent with consistency checking, but may also be relevant for a large automatically annotated corpus. In some cases, automatically annotated corpora may be manually checked after annotation, and modifications to the automatic annotation should be documented.

6. Specificity — *How detailed/shallow is the analysis.* To a certain extent, the specificity of the analysis may be shown by the levels of annotation that have been applied. However, more detailed documentation may be necessary in order to make clear the level to which an annotation is undertaken — for example some aspects of a deep or logical grammar may be included in an annotation, while others are not marked (e.g. marking of 'logical subject/object', but no marking of 'traces').

7. Ambiguity — *To what extent and in what respects has disambiguation (of machine-generated ambiguities) been carried out.* During the annotation of a corpus, all ambiguities may be resolved, or ambiguous structures may be left in the markup. Resolution of problematic ambiguities should be documented, as should any ambiguities that are left in the corpus (see section 5.2).

8. Incompleteness — *To what extent and in what respects is annotation at any particular layer incomplete.* At any particular level of annotation, certain markings may be ignored by the annotation scheme, for ease of automated annotation, or because of the intended purpose of the resource. This information should also be included in the documentation.

In the future, a further component of documentation will presumably be necessary. Assuming that there will eventually be acceptance of EAGLES guidelines as a standard for syntactic annotation, it will be highly desirable to state to what extent and in what respects a given annotation scheme conforms to the EAGLES standard. We reiterate, however, that at the present stage, the guidelines put forward in this document are highly provisional.

7 Bibliography

Aarts, J. (1992). 'Comments on A New Corpus of English: ICE.' In: Svartvik (ed.), 180–183.
Balkan, L., S. Meijer, D. Arnold, D. Estival, K. Falkedal, S. Lehmann, E. Dauphin, S. Régnier-Prost, K. Netter, S. Oepen. *Test Suite Design Annotation Scheme*, Report to LRE 62-089 (D-WP2.2), Test Suites for Natural Language Processing, University of Essex, September 1994.
Covington, M. (1990) 'A Dependency Parser for variable-word-order languages.' In: H.U. Brown (ed.), *Computer Assisted Analysis and Modelling on the IBM 3090*. Cambridge, MA : MIT Press.
Covington, M. (1991). 'Parsing discontinuous constituents in Dependency Grammar.' *Computational Linguistics* 16, 4, 234–236.
EAGLES (1996a) 'Morphosyntactic Annotation.' EAGLES document EAG–CSG/IR-T3.1. Pisa: Consiglio Nazionale delle Ricerche. Istituto di Linguistica Computazionale.
EAGLES (1996b) 'Syntactic Annotation: Survey of Annotation practices'. EAG–TCWG–SASG/2. Pisa: Consiglio Nazionale delle Ricerche. Istituto di Linguistica Computazionale.
EAGLES (1996c) 'Detailed overview of existing annotation practices'. EAG–TCWG–SASG/3. Pisa: Consiglio Nazionale delle Ricerche. Istituto di Linguistica Computazionale.
Fraser, N.M. and R.A. Hudson (1992). 'Inheritance in Word Grammar.' *Computational Linguistics* 18,2, 133–158.
Greenbaum, S. (1992) 'A New Corpus of English: ICE.' In: Svartvik (ed.), 171–179.
Halteren, H. van and N. Oostdijk (1993). 'Towards a Syntactic Database: the TOSCA Analysis System.' In J. Aarts, P. de Haan and N. Oostdijk (eds), *English Language Corpora: Design, Analysis and Exploitation*. Amsterdam: Rodopi.
Karlsson, F., A. Voutilainen, J. Heikkilä and A. Antilla (1995), *Constraint Grammar: a Language-independent System for Parsing Unrestricted Text*. Berlin and New York: Mouton de Gruyter.
Langé, Jean-Marc, F. Lemaître and A. Papin (1994). 'Corpus bilingue EN-FR annoté. Notes sur les classes grammaticales et syntaxiques.' Paper, IBM Paris.
Marcus, M., M.A. Marcinkiewicz and B. Santorini (1993). 'Building a Large Annotated Corpus of English: The Penn Treebank.' *Computational Linguistics* 19, 2, 313–330.
Marcus, M., G. Kim, M.A. Marcinkiewicz, R. MacIntyre, M. Bies, M. Ferguson, K. Katz and B. Schasberger (1994). 'The Penn Treebank: Annotating Predicate Argument Structure.' In: *Proceedings of the Human Language Technology Workshop*. San Francisco: Morgan Kaufmann Publishers.
McCord, M. (1990). 'A System for Simpler Construction of Practical Natural Language Grammars.' In: R. Studer (ed.), *Natural Language and Logic. Lecture Notes in Artificial Intelligence 459*. London and New York: Longman.
Oostdijk, N. (1991). *Corpus linguistics and the Automatic Analysis of English.* Amsterdam: Rodopi.
Radford, A. (1988). *Transformational Grammar: A first course.* Cambridge: Cambridge University Press.
Sampson, G. (1995). *English for the Computer. The* SUSANNE *Corpus and Analytic Scheme*. Oxford: Clarendon Press.
Souter, C. (1989). *A Short Handbook to the Polytechnic of Wales Corpus.* ICAME, Norwegian Computing Centre for the Humanities, P.O. Box 53, Bergen University, N-5027 Bergen, Norway.
Sperberg-McQueen and Burnard (eds) (1994). *Guidelines for electronic text encoding and interchange (*TEI *P3).* Association for Computers and the Humanities/Association for Computational Linguistics/Association for Literary and Linguistic Computing. Chicago and Oxford.
Svartvik, J. (ed.) (1992). *Directions in Corpus Linguistics*. Proceedings of Nobel Symposium 82, Stockholm. Berlin and New York: Mouton de Gruyter.
Vilkuna, M. (1989). *Free Word Order in Finnish*. Helsinki: Suomalaisen Kirjallisuuden Seura.
Voutilainen, A. (1994). *Three studies of grammar-based surface parsing of unrestricted English text.* Publication No. 24, Department of General Linguistics, University of Helsinki.
Voutilainen, A., J. Heikkilä and A. Anttila (1992). *Constraint Grammar of English. A performance-oriented Introduction.* Publication No. 21, Department of Linguistics, University of Helsinki.

8 Appendix: abbreviations

AMAZON AutoMatische ZinsONtleding ('Automatic sentence analysis')

AGFL Affix Grammar over Finite Lattices

APHB American Printing House for the Blind corpus

BNC British National Corpus

CCPP Computer Corpus Pilot Project

CGEL Comprehensive Grammar of the English Language

CLAWS Constituent Likelihood Automatic Word-tagging System

EAGLES Expert Advisory Group for Language Engineering Standards

ENGCG ENGlish Constraint Grammar

ENGTWOL ENGlish TWO-Level morphological analysis

GCE A Grammar of Contemporary English

ICAME International Computer Archive of Modern English

ICE International Corpus of English

LDC Linguistic Data Consortium

LOB London–Oslo–Bergen corpus

MT Machine Translation

POS Part Of Speech

POW Polytechnic of Wales Corpus

SASG Syntactic Annotation SubGroup

SEC Spoken English Corpus

SFG Systemic Functional Grammar

SGML Standard Generalized Mark-up Language

SUSANNE Surface and Underlying Structure ANalysis of Natural English

TCWG Text Corpus Working Group

TEI Text Encoding Initiative

TOSCA TOols for Syntactic Corpus Analysis

UCREL Unit for Computer Research of the English Language