7 Lexical Functional Grammar

Lexical Functional Grammar (LFG) was developed in the 80s by Joan Bresnan and Ron Kaplan (Bresnan & Kaplan 1982). LFG forms part of so-called West-Coast linguistics: unlike MIT, where Chomsky works and teaches, the institutes of researchers such as Joan Bresnan and Ron Kaplan are on the west coast of the USA (Joan Bresnan in Stanford and Ron Kaplan at Xerox in Palo Alto and now at the language technology firm Nuance Communications in the Bay Area in California).

Bresnan & Kaplan (1982) view LFG explicitly as a psycholinguistically plausible alternative to transformation-based approaches. For a discussion of the requirements regarding the psycholinguistic plausibility of linguistics theories, see Chapter 15.

The more in-depth works on German are Berman (1996, 2003a) and Cook (2001).

LFG has well-designed formal foundations (Kaplan & Bresnan 1982; Kaplan 1995), and hence first implementations were available rather quickly (Frey & Reyle 1983a,b; Yasukawa 1984; Block & Hunze 1986; Eisele & Dorre 1986; Kohl 1992; Kohl & Momma 1992; Wada & Asher 1986; Delmonte 1990; Her, Higinbotham & Pentheroudakis 1991; Kaplan & Maxwell III 1996; Mayo 1997, 1999; Boullier & Sagot 2005a,b; Clément 2009; Clément & Kinyon 2001).

The following is a list of languages with implemented LFG fragments, probably incomplete:

- Arabic (Attia 2008),
- Arrernte (Dras, Lareau, Börschinger, Dale, Motazedi, Rambow, Turpin & Ulinski 2012),
- Bengali (Sengupta & Chaudhuri 1997),
- Danish (Ørsnes 2002; Ørsnes & Wedekind 2003, 2004),
- English (Her, Higinbotham & Pentheroudakis 1991; Butt, Dipper, Frank & King 1999a; Riezler, King, Kaplan, Crouch, Maxwell III & Johnson 2002; King & Maxwell III 2007),
- French (Zweigenbaum 1991; Frank 1996; Frank & Zaenen 2002; Butt, Dipper, Frank & King 1999a; Clément & Kinyon 2001; Boullier, Sagot & Clément 2005; Schwarze & de Alencar 2015),
- Georgian (Meurer 2009),
- German (Rohrer 1996; Berman 1996; Kuhn & Rohrer 1997; Butt et al. 1999a; Dipper 2003; Rohrer & Forst 2006; Forst 2006; Frank 2006; Forst & Rohrer 2009),

- Hungarian (Laczkó et al. 2010),
- Indonesian (Arka, Andrews, Dalrymple, Mistica & Simpson 2009),
- Italian (Delmonte 1990; Mayo 1999; Quaglia 2014),
- Irish (Sulger 2009, 2010),
- Japanese (Her, Higinbotham & Pentheroudakis 1991; Masuichi & Ohkuma 2003; Umemoto 2006),
- Korean (Her, Higinbotham & Pentheroudakis 1991),
- Malagasy (Randriamasimanana 2006; Dalrymple, Liakata & Mackie 2006),
- Mandarin Chinese (Her, Higinbotham & Pentheroudakis 1991; Fang & King 2007),
- Murrinh-Patha (Seiss & Nordlinger 2012),
- Norwegian (Dyvik, Meurer & Rosén 2005),
- Polish (Patejuk & Przepiórkowski 2012),
- Portuguese (Alencar 2004, 2013),
- Spanish (Mayo 1999),
- Tigrinya (Kifle 2012),
- Turkish (Çetinoğlu & Oflazer 2006),
- Hungarian (Laczkó, Rákosi & Tóth 2010; Rákosi, Laczkó & Csernyi 2011),
- Urdu/Hindi (Butt, King & Roth 2007; Bögel, Butt & Sulger 2008),
- Welsh (Mittendorf & Sadler 2005) and
- Wolof (Dione 2014, 2013).

Many of theses grammars were developed in the ParGram consortium[1] (Butt, King, Niño & Segond 1999b; Butt, Dyvik, King, Masuichi & Rohrer 2002). Apart from these grammars there is a small fragment of Northern Sotho, which is currently being expanded (Faaß 2010).

Many of the LFG systems combine linguistically motivated grammars with a statistical component. Such a component can help to find preferred readings of a sentence first, it can increase the efficiency of processing and make the complete processing robust (for instance Kaplan et al. 2004; Riezler et al. 2002). Josef van Genabith's group in Dublin is working on the induction of LFG grammars from corpora (e. g. Johnson et al. 1999; O'Donovan et al. 2005; Cahill et al. 2005; Chrupala & van Genabith 2006; Guo et al. 2007; Cahill et al. 2008; Schluter & van Genabith 2009).

Some of the systems can be tested online:

[1] http://pargram.b.uib.no/research-groups/. 01.10.2015.

- http://iness.uib.no/xle-web/xle-web
- http://lfg-demo.computing.dcu.ie/lfgparser.html
- http://www.xlfg.org/

7.1 General remarks on the representational format

LFG assumes multiple levels of representation.[2] The most important are c-structure and f-structure. c-structure is the constituent structure and it is licensed by a phrase structure grammar. This phrase structure grammar uses $\overline{\text{X}}$ structures for languages for which this is appropriate. f-structure stands for functional structure. Functional structure contains information about the predicates involved and about the grammatical functions (subject, object, …) which occur in a constituent. Mappings mediate between these representational levels.

7.1.1 Functional structure

In LFG, grammatical functions such as subject and object play a very important role. Unlike in all other theories discussed in this book, they are primitives of the theory. A sentence such as (1a) will be assigned a functional structure as in (1b):

(1) a. David devoured a sandwich.

b. $$\begin{bmatrix} \textsc{pred} & \text{‘DEVOUR}\langle\textsc{subj}, \textsc{obj}\rangle\text{’} \\ \textsc{subj} & \begin{bmatrix}\textsc{pred} & \text{‘DAVID’}\end{bmatrix} \\ \textsc{obj} & \begin{bmatrix}\textsc{spec} & \text{A}\\ \textsc{pred} & \text{‘SANDWICH’}\end{bmatrix}\end{bmatrix}$$

Every lexical item contributes a PRED feature with a corresponding value. The grammatical functions governed by a head (government = subcategorization) are determined in the specification of PRED.[3] Corresponding functions are called *governable grammatical functions*. Examples of this are shown in Table 7.1 on the following page (Dalrymple 2006). The PRED specification corresponds to the theta grid in GB theory. The valence of a head is specified by the PRED value.

The non-governable grammatical functions are given in Table 7.2 on the next page. Topic and focus are information-structural terms. There are a number of works on their exact definition, which differ to varying degrees (Kruijff-Korbayová & Steedman 2003:

[2] The English examples and their analyses discussed in this section are taken from Dalrymple (2001) and Dalrymple (2006).

[3] In the structure in (1b), the SUBJ and OBJ in the list following *devour* are identical to the values of SUBJ and OBJ in the structure. For reasons of presentation, this will not be explicitly indicated in this structure and following structures.

Table 7.1: Governable grammatical functions

SUBJ:	subject
OBJ:	object
COMP:	sentential complement or closed (non-predicative) infinitival complement
XCOMP:	open (predicative) complement, often infinitival, the SUBJ function is externally controlled
OBJ_θ:	secondary OBJ functions that are related to a special, language specific set of grammatical roles; English has OBJ_{THEME} only.
OBL_θ:	a group of thematically restricted oblique functions as for instance OBL_{GOAL} or OBL_{AGENT}. These often correspond to adpositional phrases in c-structure.

Table 7.2: Non-governable grammatical functions

ADJ:	adjuncts
TOPIC:	the topic of an utterance
FOCUS:	the focus of an utterance

253–254), but broadly speaking, one can say that the focus of an utterance constitutes new information and that the topic is old or given information. Bresnan (2001: 97) uses the following question tests in order to determine topic and focus:

(2) Q: What did you name your cat?
A: Rosie I named her. (*Rosie* = FOCUS)

(3) Q: What did you name your pets?
A: My dog, I named Harold. My cat, I named Rosie. (*my dog*, *my cat* = TOPIC)

f-structures are characterized using functional descriptions, for example, one can refer to a value of the feature TENSE in the functional structure f using the following expression:

(4) (f TENSE)

It is possible to say something about the value which this feature should have in the feature description. The following descriptions express the fact that in the structure f, the feature TENSE must have the value PAST.

(5) (f TENSE) = PAST

The value of a feature may also be a specific f-structure. The expression in (6) ensures that the SUBJ feature in f is the f-structure g:

(6) $(f\ \textsc{subj}) = g$

For the analysis of (7a), we get the constraints in (7b):

(7) a. David sneezed.

b. $(f\ \textsc{pred}) = \text{‘SNEEZE}\langle\textsc{subj}\ \rangle\text{’}$
$(f\ \text{TENSE}) = \text{PAST}$
$(f\ \textsc{subj}) = g$
$(g\ \textsc{pred}) = \text{‘DAVID’}$

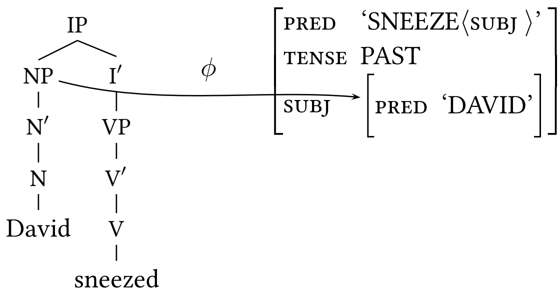
The description in (7b) describes the following structure:

(8) $$f\colon \begin{bmatrix} \textsc{pred} & \text{‘SNEEZE}\langle\textsc{subj}\ \rangle\text{’} \\ \textsc{tense} & \text{PAST} \\ \textsc{subj} & g\colon \begin{bmatrix} \textsc{pred} & \text{‘DAVID’} \end{bmatrix} \end{bmatrix}$$

But (7b) also describes many other structures which contain further features. We are only interested in minimal structures that contain the information provided in the description.

(9) shows how a node in the c-structure can be connected to the f-structure for the entire sentence:

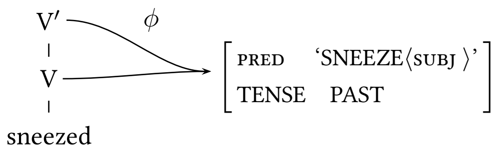
(9)



The function ϕ from the NP-node to the f-structure corresponding to the NP is depicted with an arrow marked ϕ.

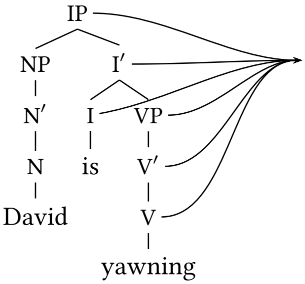
A phrase and its head always correspond to the same f-structure:

(10)



In LFG grammars of English, the CP/IP system is assumed as in GB theory (see Section 3.1.5). IP, I′ and I (and also VP) are mapped onto the same f-structure.

(11) a. David is yawning.

b. 

$$\begin{bmatrix} \textsc{pred} & \text{‘YAWN}\langle\textsc{subj}\rangle\text{’} \\ \text{TENSE} & \text{PRES} \\ \textsc{subj} & \begin{bmatrix} \textsc{pred} & \text{‘DAVID’} \end{bmatrix} \end{bmatrix}$$

f-structures have to fulfill two well-formedness conditions: they have to be both *complete* and *coherent*. Both these conditions will be discussed in the following sections.

7.1.2 Completeness

Every head adds a constraint of the PRED value of the corresponding f-structure. In determining completeness, one has to check that the elements required in the PRED value are actually realized. In (12b), OBJ is missing a value, which is why (12a) is ruled out by the theory.

(12) a. * David devoured.

b.
$$\begin{bmatrix} \textsc{pred} & \text{‘DEVOUR}\langle\textsc{subj,obj}\rangle\text{’} \\ \textsc{subj} & \begin{bmatrix} \textsc{pred} & \text{‘DAVID’} \end{bmatrix} \end{bmatrix}$$

7.1.3 Coherence

The Coherence Constraint requires that all argument functions in a given f-structure have to be selected in the value of the local PRED attribute. (13a) is ruled out because COMP does not appear under the arguments of *devour*.

(13) a. * David devoured a sandwich that Peter sleeps.

b.
$$\begin{bmatrix} \textsc{pred} & \text{‘DEVOUR}\langle\textsc{subj,obj}\rangle\text{’} \\ \textsc{subj} & [\ \textsc{pred}\ \text{‘DAVID’}\ ] \\ \textsc{obj} & \begin{bmatrix} \textsc{spec} & \text{A} \\ \textsc{pred} & \text{‘SANDWICH’} \end{bmatrix} \\ \textsc{comp} & \begin{bmatrix} \textsc{pred} & \text{‘SLEEP}\langle\textsc{subj}\rangle\text{’} \\ \textsc{subj} & \begin{bmatrix} \textsc{pred} & \text{‘PETER’} \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

The constraints on completeness and coherence together ensure that only those arguments required in the PRED specification are actually realized. Both of those constraints taken together correspond to the Theta Criterion in GB theory (see page 94).[4]

7.1.4 Restrictions on the c-structure/f-structure relation

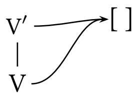
Symbols in c-structures are assigned restrictions for f-structures. The following symbols are used: '↑' refers to the f-structure of the immediately dominating node and '↓' refers to the f-structure of the c-structure node bearing the annotation. A common annotation is '↑ = ↓'. This constraint states that the f-structure of the mother node is identical to that of the annotated category:

(14) $$\begin{array}{ll} \mathrm{V}' \rightarrow & \mathrm{V} \\ & \uparrow = \downarrow \\ & \text{f-structure of the mother = own f-structure} \end{array}$$

The annotation '↑ = ↓' is below the head of a structure.

Phrases which are licensed by the annotated c-structure in (14) can be visualized as follows:

(15)

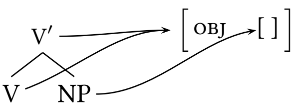


(16) shows a V′ rule with an object:

(16) $$\begin{array}{llll} \mathrm{V}' & \rightarrow & \mathrm{V} & \mathrm{NP} \\ & & \uparrow = \downarrow & (\uparrow \textsc{obj}) = \downarrow \end{array}$$

The annotation on the NP signals that the OBJ value in the f-structure of the mother (↑ OBJ) is identical to the f-structure of the NP node, that is, to everything that is contributed from the material below the NP node (↓). This is shown in the figure in (17):

(17)

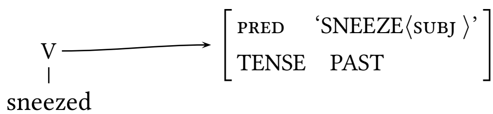


In the equation (↑ OBJ) = ↓, the arrows '↑' and '↓' correspond to feature structures. '↑' and '↓' stand for the f and g in equations such as (6).

(18) is an example with an intransitive verb and the corresponding visualization is in (19):

(18) *sneezed* V (↑ PRED) = 'SNEEZE⟨SUBJ ⟩'
(↑ TENSE) = PAST

(19)



[4] For the differences between predicate-argument structures in LFG and the Deep Structure oriented Theta Criterion, see Bresnan & Kaplan (1982: xxvi–xxviii).

7.1.5 Semantics

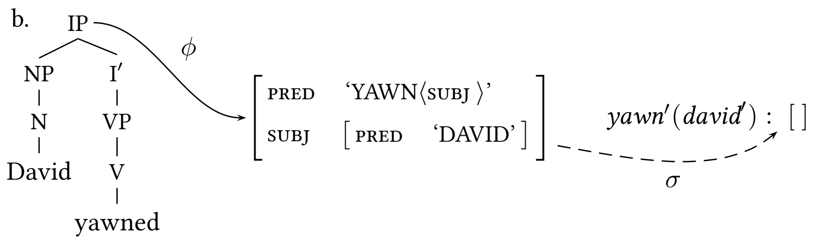
Following Dalrymple (2006: 90–92), *glue semantics* is the dominant approach to semantic interpretation in LFG (Dalrymple, Lamping & Saraswat 1993; Dalrymple 2001: Chapter 8). There are, however, other variants where Kamp's discourse representation structures (Kamp & Reyle 1993) are used (Frey & Reyle 1983a,b).

In the following, glue semantics will be presented in more detail.[5] Under a glue-based approach, it is assumed that f-structure is the level of syntactic representation which is crucial for the semantic interpretation of a phrase, that is, unlike GB theory, it is not the position of arguments in the tree which play a role in the composition of meaning, but rather functional relations such as SUBJ and OBJ. Glue semantics assumes that each substructure of the f-structure corresponds to a semantic resource connected to a meaning and furthermore, that the meaning of a given f-structure comes from the sum of these parts. The way the meaning is assembled is regulated by certain instructions for the combination of semantic resources. These instructions are given as a set of logic premises written in linear logic as *glue language*. The computation of the meaning of an utterance corresponds to a logical conclusion.

This conclusion is reached on the basis of logical premises contributed by the words in an expression or possibly even by a syntactic construction itself. The requirements on how the meaning of the parts can be combined to yield the full meaning are expressed in linear logic, a resource-based logic. Linear logic is different from classic logic in that it does not allow that premises of conclusions are not used at all or more than once in a derivation. Hence, in linear logic, premises are resources which have to be used. This corresponds directly to the use of words in an expression: words contribute to the entire meaning exactly once. It is not possible to ignore them or to use their meaning more than once. A sentence such as *Peter knocked twice.* does not mean the same as *Peter knocked.* The meaning of *twice* must be included in the full meaning of the sentences. Similarly, the sentence cannot mean the same as *Peter knocked twice twice.*, since the semantic contribution of a given word cannot be used twice.

The syntactic structure for the sentence in (20a) together with its semantic representation is given in (20b):

(20) a. David yawned.

b. 

The semantic structure of this sentence is connected to the f-structure via the correspon-

[5] The following discussion heavily draws from the corresponding section of Dalrymple (2006). (It is a translation of my translation of the original material into German.)

dence function σ (depicted here as a dashed line). The semantic representation is derived from the following lexical information for the verb *yawned*:

(21) $\lambda x.yawn'(x) : (\uparrow \textsc{subj})_\sigma \multimap \uparrow_\sigma$

This formula is referred to as the *meaning constructor*. Its job is to combine the meaning of *yawned* – a one place predicate $\lambda x.yawn'(x)$ – with the formula $(\uparrow \textsc{subj})_\sigma \multimap \uparrow_\sigma$ in linear logic. Here, the connective $\multimap$ is the *linear implication* symbol of linear logic. The symbol contains the meaning that *if* a semantic resource $(\uparrow \textsc{subj})_\sigma$ for the meaning of the subject is available, *then* a semantic resource for $\uparrow_\sigma$ must be created which will stand for the entire meaning of the sentence. Unlike the implication operator of classic logic, the linear implication must consume and produce semantic resources: the formula $(\uparrow \textsc{subj})_\sigma \multimap \uparrow_\sigma$ states that if a semantic resource $(\uparrow \textsc{subj})_\sigma$ is found, it is consumed and the semantic resource $\uparrow_\sigma$ is produced.

Furthermore, it is assumed that a proper name such as *David* contributes its own semantic structure as a semantic resource. In an utterance such as *David yawned*, this resource is consumed by the verb *yawned*, which requires a resource for its SUBJ in order to produce the resource for the entire sentence. This corresponds to the intuition that a verb in any given sentence requires the meaning of its arguments in order for the entire sentence to be understood.

The f-structure of *David yawned* with the instantiated meaning construction contributed by *David* and *yawned* is given in (22):

(22)
$$y : \begin{bmatrix} \textsc{pred} & \text{‘YAWN}\langle\textsc{subj}\,\rangle\text{’} \\ \textsc{subj} & d : \begin{bmatrix} \textsc{pred} & \text{‘DAVID’} \end{bmatrix} \end{bmatrix}$$

[David] $david' : d_\sigma$

[yawn] $\lambda x.yawn'(x) : d_\sigma \multimap y_\sigma$

The left side of the meaning constructor marked by **[David]** is the meaning of the proper name *David*, *david′* to be precise. The left-hand side of the meaning constructor **[yawn]** is the meaning of the intransitive verb – a one-place predicate $\lambda x.yawn'(x)$.

Furthermore, one must still postulate further rules to determine the exact relation between the right-hand side (the glue) of the meaning constructors in (22) and the left-hand side (the meaning). For simple, non-implicational meaning constructors such as **[David]** in (22), the meaning on the left is the same as the meaning of the semantic structure on the right. Meaning constructors such as **[yawn]** have a λ-expression on the left, which has to be combined with another expression via functional application (see Section 2.3). The linear implication on the right-hand side must be applied in parallel. This combined process is shown in (23).

(23)
$$\frac{\begin{array}{c} x : f_\sigma \\ P : f_\sigma \multimap g_\sigma \end{array}}{P(x) : g_\sigma}$$

The right-hand side of the rule corresponds to a logical conclusion following the *modus ponens* rule. With these correspondences between expressions in linear logic and the meanings themselves, we can proceed as shown in (24), which is based on Dalrymple (2006: 92):

(24)

$\mathit{david}' : d_\sigma$	The meaning *david′* is associated with the semantic structure of SUBJ d_σ.
$\lambda x.\mathit{yawn}'(x) : d_\sigma \multimap y_\sigma$	If we find the semantic resource for the SUBJ d_σ on the glue side, this resource is consumed and the semantic resource for the entire sentence y_σ is produced. On the meaning side, we apply the function $\lambda x.\mathit{yawn}'(x)$ to the meaning associated with d_σ.
$\mathit{yawn}'(\mathit{david}') : y_\sigma$	We have created the semantic structure y_σ for the entire sentence, associated with the meaning of *yawn′*(*david′*).

After combining the respective meanings of *yawned* and *David* and then carrying out β-reduction, we arrive at the desired result of $\mathit{yawn}'(\mathit{david}')$ as the meaning of *David yawned.*

Glue analyses of quantification, modification and other phenomena have been investigated in a volume on glue semantics (Dalrymple 1999). Particularly problematic for these approaches are cases where there appear to be too many or too few resources for the production of utterances. These kinds of cases have been discussed by Asudeh (2004).

7.1.6 Adjuncts

Adjuncts are not selected by their head. The grammatical function ADJ is a non-governable grammatical function. Unlike arguments, where every grammatical function can only be realized once, a sentence can contain multiple adjuncts. The value of ADJ in the f-structure is therefore not a simple structure as with the other grammatical functions, but rather a set. For example, the f structure for the sentence in (25a) contains an ADJ set with two elements: one for *yesterday* and one for *at noon.*

(25) a. David devoured a sandwich at noon yesterday.

b. $$\begin{bmatrix} \textsc{pred} & \text{‘DEVOUR}\langle\textsc{subj},\textsc{obj}\rangle\text{’} \\ \textsc{subj} & \begin{bmatrix} \textsc{pred} & \text{‘DAVID’} \end{bmatrix} \\ \textsc{obj} & \begin{bmatrix} \textsc{spec} & \text{A} \\ \textsc{pred} & \text{‘SANDWICH’} \end{bmatrix} \\ \textsc{adj} & \left\{ \begin{bmatrix} \textsc{pred} & \text{‘YESTERDAY’} \end{bmatrix}, \begin{bmatrix} \textsc{pred} & \text{‘AT}\langle\textsc{obj}\ \rangle\text{’} \\ \textsc{obj} & \begin{bmatrix} \textsc{pred} & \text{‘NOON’} \end{bmatrix} \end{bmatrix} \right\} \end{bmatrix}$$

The annotation on the c-structure rule for adjuncts requires that the f-structure of the adjuncts be part of the ADJ set of the mother's f-structure:

(26) $$\begin{array}{llll} \text{V}' & \rightarrow & \text{V}' & \text{PP} \\ & & \uparrow = \downarrow & \downarrow \in (\uparrow \textsc{adj}) \end{array}$$

The representation of adjuncts in a set is not sufficient to characterize the meaning of an utterance containing scope-bearing adjuncts (as for instance the negation in sentences like (31) on page 107). In order to determine scopal relations, one has to refer to the linear order of the adjuncts, that is, the c-structure. For linearization restrictions in LFG, see Zaenen & Kaplan (1995).

7.2 Passive

Bresnan & Mchombo (1995) argue that one should view words as "atoms" of which syntactic structure is comprised (*lexical integrity*[6]).

Syntactic rules cannot create new words or make reference to the internal structure of words. Every terminal node (each "leaf" of the tree) is a word. It follows from this that analyses such as the GB analysis of Pollock (1989) in Figure 7.1 on the following page for the French example in (27) are ruled out (the figure is taken from Kuhn (2007: 617)):

(27) Marie ne parlerait pas
Marie NEG speak.COND.3SG NEG
'Marie would not speak.'

In Pollock's analysis, the various morphemes are in specific positions in the tree and are combined only after certain movements have been carried out.

The assumption of lexical integrity is made by all theories discussed in this book with the exception of GB and Minimalism. However, formally, this is not a must as it is also possible to connect morphemes to complex syntactic structures in theories such as Categorial Grammar, GPSG, HPSG, CxG, DG and TAG. As far as I know, this kind of analysis has never been proposed.

[6] See Anderson (1992: 84) for more on lexical integrity.

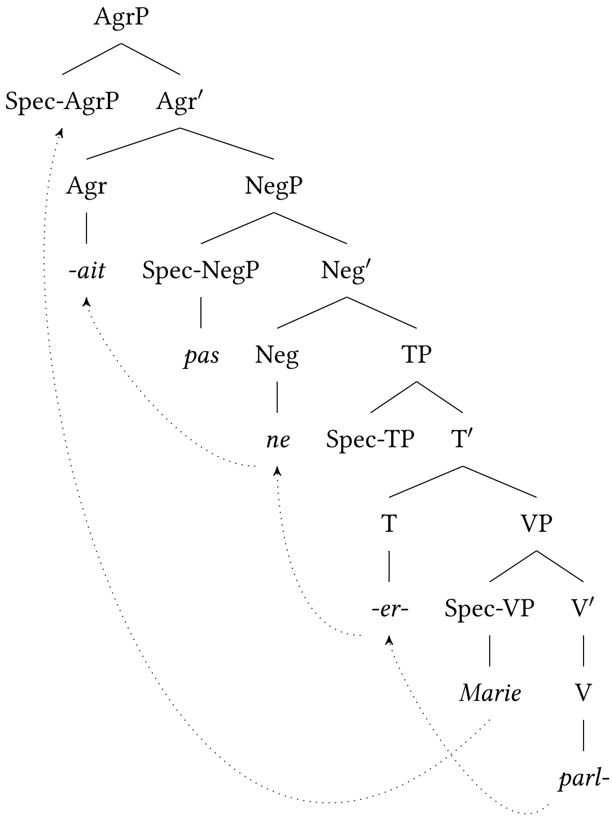


Figure 7.1: Pollock's analysis of *Marie ne parlerait pas* 'Marie would not speak.' according to Kuhn (2007: 617)

Bresnan noticed that, as well as passivized verbs, there are passivized adjectives which show the same morphological idiosyncrasies as the corresponding participles (Bresnan 1982c: 21; Bresnan 2001: 31). Some examples are given in (28):

(28) a. a well-written novel (write – written)
  b. a recently given talk (give – given)
  c. my broken heart (break – broken)
  d. an uninhabited island (inhabit – inhabited)
  e. split wood (split – split)

If one assumes lexical integrity, then adjectives would have to be derived in the lexicon. If the verbal passive were not a lexical process, but rather a phrase-structural one, then the form identity would remain unexplained.

In LFG, grammatical functions are primitives, that is, they are not derived from a position in the tree (e. g. Subject = SpecIP). Words (fully inflected word-forms) determine the grammatical function of their arguments. Furthermore, there is a hierarchy of grammatical functions. During participle formation in morphology, the highest verbal argument is suppressed. The next highest argument moves up and is not realized as the OBJECT but rather as the SUBJECT. This was explicitly encoded in earlier work (Bresnan 1982c: 8):

(29) Passivization rule:
$(\textsc{subj}) \mapsto \varnothing/(\mathrm{OBL})$
$(\textsc{obj}) \mapsto (\textsc{subj})$

The first rule states that the subject is either not realized (∅) or it is realized as an oblique element (the *by*-PP in English). The second rule states that if there is an accusative object, this becomes the subject.

In later work, the assignment of grammatical functions was taken over by Lexical Mapping Theory (Bresnan & Kanerva 1989). It is assumed that thematic roles are ordered in a universally valid hierarchy (Bresnan & Kanerva 1989; Bresnan 2001: 307): agent > beneficiary > experiencer/goal > instrument > patient/theme > locative. Patient-like roles are marked as unrestricted ([−r]) in a corresponding representation, the so-called a-structure. Secondary patient-like roles are marked as *objective*, ([+o]) and all other roles are marked as non-objective ([−o]). For the transitive verb *schlagen* 'to beat', we have the following:

(30)

		Agent	Patient
a-structure	*schlagen* 'beat'	⟨ x	y ⟩
		[−o]	[−r]

The mapping of a-structure to f-structure is governed by the following restrictions:

(31) a. Subject-Mapping-Principle: The most prominent role marked with [−o] is mapped to SUBJ if it is initial in the a-structure. Otherwise, the role marked with [−r] is mapped to SUBJ.

b. The argument roles are connected to grammatical functions as shown in the following table. Non-specified values for o and r are to be understood as '+':

	[−r]	[+r]
[−o]	SUBJ	OBL_θ
[+o]	OBJ	OBJ_θ

c. *Function-Argument Biuniqueness*: Every a-structure role must be associated to exactly one function and vice versa.

For the argument structure in (30), the principle in (31a) ensures that the agent x receives the grammatical function SUBJ. (31b) adds an o-feature with the value '+' so that the patient y is associated with OBJ:

(32)
$$\begin{array}{llll} & & \text{Agent} & \text{Patient} \\ \text{a-structure} & \textit{schlagen}\ \text{`beat'} & \langle\ \text{x} & \text{y}\ \ \rangle \\ & & [-\text{o}] & [-\text{r}] \\ \cline{3-4} & & \text{SUBJ} & \text{OBJ} \end{array}$$

Under passivization, the most prominent role is suppressed so that only the [−r] marked patient role remains. Following (31a), this role will then be mapped to the subject.

(33)
$$\begin{array}{llll} & & \text{Agent} & \text{Patient} \\ \text{a-structure} & \textit{schlagen}\ \text{`beat'} & \langle\ \text{x} & \text{y}\ \ \rangle \\ & & [-\text{o}] & [-\text{r}] \\ \cline{3-4} & & \varnothing & \text{SUBJ} \end{array}$$

Unlike the objects of transitive verbs, the objects of verbs such as *helfen* 'help' are marked as [+o] (Berman 1999). The lexical case of the objects is given in the a-structure, since this case (dative) is linked to a semantic role (Zaenen, Maling & Thráinsson 1985: 465). The corresponding semantic roles are obligatorily mapped to the grammatical function OBJ_θ.

(34)
$$\begin{array}{llll} & & \text{Agent} & \text{Beneficiary} \\ \text{a-structure} & \textit{helfen}\ \text{`help'} & \langle\ \text{x} & \text{y}\ \ \rangle \\ & & [-\text{o}] & [+\text{o}]/\text{DAT} \\ \cline{3-4} & & \text{SUBJ} & \text{OBJ}_\theta \end{array}$$

Passivization will yield the following:

(35)
$$\begin{array}{llll} & & \text{Agent} & \text{Beneficiary} \\ \text{a-structure} & \textit{helfen}\ \text{`help'} & \langle\ \text{x} & \text{y}\ \ \rangle \\ & & [-\text{o}] & [+\text{o}]/\text{DAT} \\ \cline{3-4} & & \varnothing & \text{OBJ}_\theta \end{array}$$

Since there is neither a [−o] nor a [−r] argument, no argument is connected to the subject function. The result is an association of arguments and grammatical functions, which corresponds to that found in impersonal passives.

These mapping principles may seem complex at first glance, but they play a role in analyzing an entire range of phenomena, e. g. the analysis of unaccusative verbs (Bresnan & Zaenen 1990). For the analysis of the passive, we can now say that the passive suppresses the highest [−o] role. Any mention of some other possibly present object in the passive rule is no longer necessary.

7.3 Verb position

There are two possibilities for the analysis of verb placement in German.

- a trace in verb-final position (as in GB) (see Choi 1999, Berman 1996: Section 2.1.4) and

- so-called *extended head domains* (see Berman 2003a).

In the analysis of extended head domains, the verb is simply omitted from the verb phrase. The following preliminary variant of the VP rule is used:[7]

(36) VP → (NP) (NP) (NP) (V)

All components of the VP are optional as indicated by the brackets. As in GB analyses, the verb in verb-first clauses is in C. No I projection is assumed – as in a number of GB works (Haider 1993, 1995, 1997a; Sternefeld 2006: Section IV.3), since it is difficult to motivate its existence for German (Berman 2003a: Section 3.2.2). The verb contributes its f-structure information from the C position. Figure 7.2 contains a simplified version of the analysis proposed by Berman (2003a: 41).

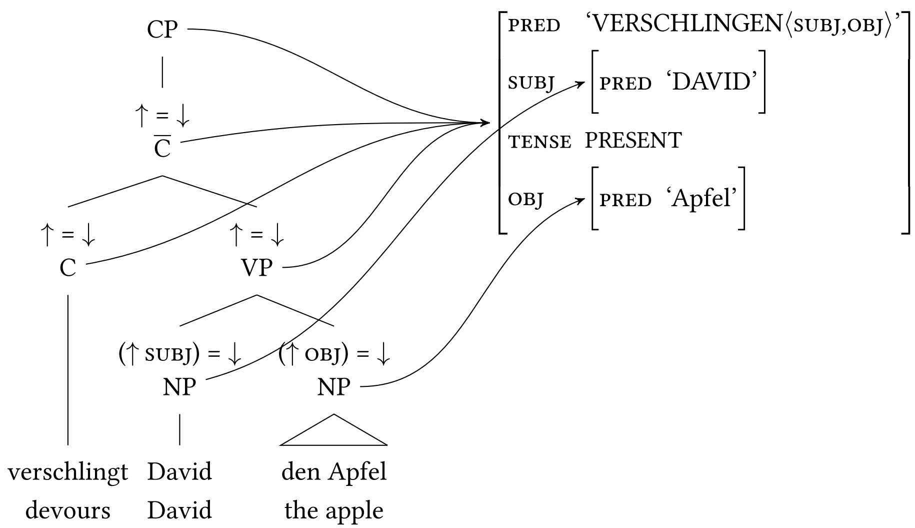


Figure 7.2: Analysis of verb placement following Berman (2003a: 41)

After what we learned about phrase structure rules in Chapters 2 and 5, it may seem strange to allow VPs without V. This is not a problem in LFG, however, since for the analysis of a given sentence, it only has to be ensured that all the necessary parts (and only these) are present. This is ensured by the constraints on completeness and coherence. Where exactly the information comes from is not important. In Figure 7.2, the verb information does not come from the VP, but rather from the C node. C′ is licensed by a special rule:

(37) $C' \rightarrow \underset{\uparrow = \downarrow}{C} \quad \underset{\uparrow = \downarrow}{VP}$

[7] See Bresnan (2001: 110) and Dalrymple (2006: Section 2.2) for a corresponding rule with optional constituents on the right-hand side of the rule.

In LFG rules, there is normally only one element annotated with '↑ = ↓', namely the head. In (37), there are two such elements, which is why both equally contribute to the f-structure of the mother. The head domain of V has been extended to C. The information about SUBJ and OBJ comes from the VP and the information about PRED from C.

7.4 Local reordering

Two possibilities for treating local reordering have been discussed in the literature:[8]

- movement of arguments from a base configuration as in GB (see Choi 1999)
- direct licensing by phrase structure rules (see Berman: 1996: Section 2.1.3.1; 2003a)

If one assumes that traces are relevant for the semantic interpretation of a given structure, then the first option has the same problems as movement-based GB analyses. These have already been discussed in Section 3.5.

In what follows, I will present the analysis proposed by Berman (1996: Section 2.1.3) in a somewhat simplified form. Case and grammatical functions of verbal arguments are determined in the lexicon (Berman 1996: 22). (38) shows the lexical entry for the verb

[8] Kaplan (1995: 20–21) shows how one can write grammars in the ID/LP format in LFG. A GPSG-like analysis of German constituent order has not been proposed in the LFG framework.

verschlingen 'devour':[9,10]

(38) *verschlingt* V (↑ PRED) = 'VERSCHLINGEN⟨SUBJ, OBJ ⟩'
(↑ SUBJ AGR CAS) = NOM
(↑ OBJ AGR CAS) = ACC
(↑ TENSE) = PRESENT

Berman proposes an analysis that does not combine the verb with all its arguments and adjuncts at the same time, as was the case in GPSG. Instead, she chooses the other extreme and assumes that the verb is not combined with an adjunct or an argument, but rather forms a VP directly. The rule for this is shown in (39):

(39) VP → (V)
↑ = ↓

At first sight, this may seem odd since a V such as *verschlingen* 'devour' does not have the same distribution as a verb with its arguments. However, one should recall that the

[9] The four cases in German can be represented using two binary features (GOV, OBL) (Berman 1996: 22). Nominative corresponds to GOV− and OBL− and accusative to GOV+ and OBL−. This kind of encoding allows one to leave case partially underspecified. If one does not provided a value for GOV, then an element with OBL− is compatible with both nominative and accusative. Since this underspecification is not needed in the following discussion, I will omit this feature decomposition and insert the case values directly.

[10] Alternative analyses derive the grammatical function of an NP from its case (Berman (2003a: 37) for German; Bresnan (2001: 187, 201) for German and Russian).

(i) (↓ CASE) = ACC ⇒ (↑ OBJ) = ↓

Karttunen (1989: Section 2.1) makes a similar suggestion for Finnish in the framework of Categorial Grammar. Such analyses are not entirely unproblematic as case cannot always be reliably paired with grammatical functions. In German, as well as temporal accusatives (ii.a), there are also verbs with two accusative objects (ii.b–c) and predicative accusatives (ii.d).

(ii) a. Er arbeitete den ganzen Tag.
he worked the.ACC whole.ACC day

b. Er lehrte ihn den Ententanz.
he taught him.ACC the.ACC duck.dance

c. Das kostet ihn einen Taler.
that costs him.ACC a.ACC taler

d. Sie nannte ihn einen Lügner.
she called him.ACC a.ACC liar

All of these accusatives can occur in long-distance dependencies (see Section 7.5):

(iii) Wen glaubst du, dass ich getroffen habe.
who believe you that I met have
'Who do you think I met?'

wen is not the object of *glauben* 'believe' and as such cannot be included in the f-structure of *glauben* 'believe'. One would have to reformulate the implication in (i) as a disjunction of all possible grammatical functions of the accusative and in addition account for the fact that accusatives can come from a more deeply embedded f-structure.

constraints pertaining to coherence and completeness of f-structures play an important role so that the theory cannot make incorrect predictions.

Since the verb can occur in initial position, it is marked as optional in the rule in (39) (see Section 7.3).

The following rule can be used additionally to combine the verb with its subject or object.

(40) $\begin{array}{lll} \text{VP} \rightarrow & \text{NP} & \text{VP} \\ & (\uparrow \textsc{subj} \mid \textsc{obj} \mid \text{OBJ}_{\theta}) = \downarrow & \uparrow = \downarrow \end{array}$

The '|' here stands for a disjunction, that is, the NP can either be the subject or the object of the superordinate f-structure. Since VP occurs both on the left and right-hand side of the rule in (40), it can be applied multiple times. The rule is not complete, however. For instance, one has to account for prepositional objects, for clausal arguments, for adjectival arguments and for adjuncts. See footnote 12 on page 240.

Figure 7.3 shows the analysis for (41a).

(41) a. [dass] David den Apfel verschlingt
that David the apple devours
'that David is devouring the apple'

b. [dass] den Apfel David verschlingt
that the apple David devours

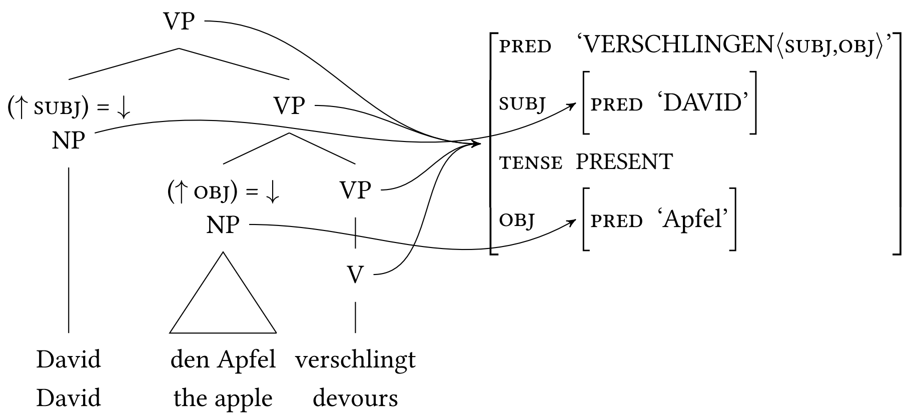


Figure 7.3: Analysis of SOV order following Berman (1996)

The analysis of (41b) is shown in Figure 7.4 on the next page. The analysis of (41b) differs from the one of (41a) only in the order of the replacement of the NP node by the subject or object.

One further fact must be discussed: in the rule (39), the verb is optional. If it is omitted, the VP is empty. In this way, the VP rule in (40) can have an empty VP on the right-hand

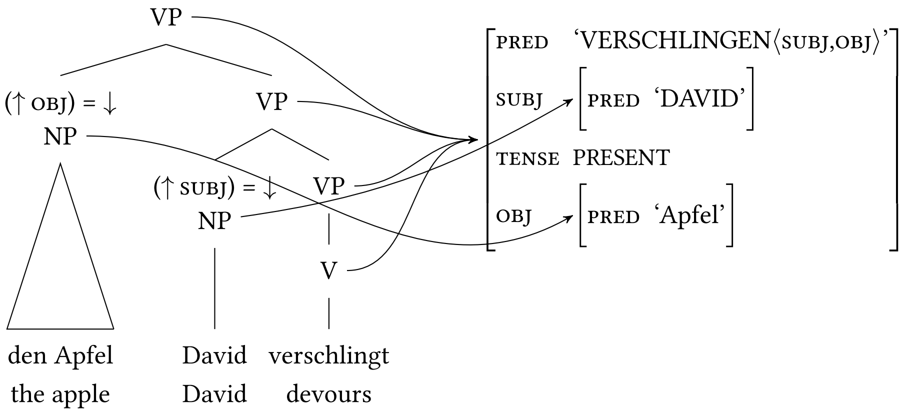


Figure 7.4: Analysis of OSV order following Berman (1996)

side of the rule. This VP is also simply omitted even though the VP symbol in the right-hand side of rule (40) is not marked as optional. That is, the corresponding symbol then also becomes optional as a result of taking the rest of the grammar into consideration as well as possible interactions with other rules.

7.5 Long-distance dependencies and functional uncertainty

We have seen that LFG can explain phenomena such as passivization, local reordering as well as verb placement without transformations. In Chapter 5 on GPSG, we already saw that the development of a transformation-less analysis for long-distance dependencies constitutes a real achievement. In LFG, Kaplan & Zaenen (1989) proposed another transformation-less analysis of long-distance dependencies, which we will consider in further detail in what follows.

In example (42), the displaced constituent *Chris* is characterized by two functions:

(42) Chris, we think that David saw.

For one, it has an argument function which is normally realized in a different position (the OBJ function of *saw* in the above example) and additionally it has a discourse function: a certain emphasis of the information-structural status in this construction (TOPIC in the matrix clause). In LFG, TOPIC and FOCUS are assumed to be grammaticalized discourse functions (furthermore, SUBJ is classified as the default discourse function). Only grammaticalized discourse functions are represented on the level of f-structure, that is, those that are created by a fixed syntactic mechanism and that interact with the rest of the syntax.

Unlike argument functions, the discourse functions TOPIC and FOCUS are not lexically subcategorized and are therefore not subject to the completeness and coherence conditions. The values of discourse function features like TOPIC and FOCUS are identified with an f-structure that bears an argument function. (43) gives the f-structure for the sentence in (42):

(43)
$$\begin{bmatrix} \text{PRED} & \text{`THINK}\langle \text{ SUBJ, COMP } \rangle\text{'} \\ \text{TOPIC} & \begin{bmatrix} \text{PRED} & \text{`CHRIS'} \end{bmatrix} \\ \text{SUBJ} & \begin{bmatrix} \text{PRED} & \text{`pro'} \end{bmatrix} \\ \text{COMP} & \begin{bmatrix} \text{PRED} & \text{`SEE}\langle \text{ SUBJ, OBJ}\rangle\text{'} \\ \text{SUBJ} & \begin{bmatrix} \text{PRED} & \text{`DAVID'} \end{bmatrix} \\ \text{OBJ} & \end{bmatrix} \end{bmatrix}$$

The connecting line means that the value of TOPIC is identical to the value of COMP|OBJ. In Chapter 6 on feature descriptions, I used boxes for structure sharing rather than connecting lines, since boxes are more common across frameworks. It is possible to formulate the structure sharing in (43) as an f-structure constraint as in (44):

(44) $(\uparrow \text{TOPIC}) = (\uparrow \text{COMP OBJ})$

Fronting operations such as (42) are possible from various levels of embedding: for instance, (45a) shows an example with less embedding. The object is located in the same f-structure as the topic. However, the object in (42) comes from a clause embedded under *think*.

The f-structure corresponding to (45a) is given in (45b):

(45) a. Chris, we saw.

b.
$$\begin{bmatrix} \text{PRED} & \text{`SEE}\langle \text{ SUBJ, OBJ}\rangle\text{'} \\ \text{TOPIC} & \begin{bmatrix} \text{PRED} & \text{`CHRIS'} \end{bmatrix} \\ \text{SUBJ} & \begin{bmatrix} \text{PRED} & \text{`pro'} \end{bmatrix} \\ \text{OBJ} & \end{bmatrix}$$

The identity restriction for TOPIC and object can be formulated in this case as in (46):

(46) $(\uparrow \text{TOPIC}) = (\uparrow \text{OBJ})$

Example (47a) shows a case of even deeper embedding than in (42) and (47b,c) show the corresponding f-structure and the respective restriction.

(47) a. Chris, we think Anna claims that David saw.

b. $$\left[\begin{array}{ll}
\textsc{pred} & \text{`THINK}\langle\, \textsc{subj}, \textsc{comp}\,\rangle\text{'} \\
\textsc{topic} & \left[\begin{array}{ll}\textsc{pred} & \text{`CHRIS'}\end{array}\right] \\
\textsc{subj} & \left[\begin{array}{ll}\textsc{pred} & \text{`pro'}\end{array}\right] \\
\textsc{comp} & \left[\begin{array}{ll}
\textsc{pred} & \text{`CLAIM}\langle\, \textsc{subj}, \textsc{comp}\,\rangle \\
\textsc{subj} & \left[\begin{array}{ll}\textsc{pred} & \text{`ANNA'}\end{array}\right] \\
\textsc{comp} & \left[\begin{array}{ll}
\textsc{pred} & \text{`SEE}\langle\, \textsc{subj}, \textsc{obj}\rangle \\
\textsc{subj} & \left[\begin{array}{ll}\textsc{pred} & \text{`DAVID'}\end{array}\right] \\
\textsc{obj} &
\end{array}\right]
\end{array}\right]
\end{array}\right]$$

c. (↑ TOPIC) = (↑ COMP COMP OBJ)

The restrictions in (44), (46) and (47c) are c-structure constraints. The combination of a c-structure with (44) is given in (48):

(48) $$\begin{array}{llll}
\text{CP} & \rightarrow & \text{XP} & \text{C}' \\
 & & (\uparrow \textsc{topic}) = \downarrow & \uparrow = \downarrow \\
 & & (\uparrow \textsc{topic}) = (\uparrow \textsc{comp obj}) &
\end{array}$$

(48) states that the first constituent contributes to the TOPIC value in the f-structure of the mother and furthermore that this topic value has to be identical to that of the object in the complement clause. We have also seen examples of other embeddings of various depths. We therefore need restrictions of the following kind as in (49):

(49)
   a. (↑ TOPIC) = (↑ OBJ)
   b. (↑ TOPIC) = (↑ COMP OBJ)
   c. (↑ TOPIC) = (↑ COMP COMP OBJ)
   d. …

The generalization emerging from these equations is given in (50):

(50) (↑ TOPIC) = (↑ COMP* OBJ)

Here, '*' stands for an unrestricted number of occurrences of COMP. This means of leaving the possible identification of discourse and grammatical function open is known as *functional uncertainty*, see Kaplan & Zaenen (1989).

As was shown in the discussion of examples (2) and (3) on page 222, it is not the case that only a TOPIC can be placed in the specifier position of CP in English as FOCUS can occur there too. One can use disjunctions in LFG equations and express the corresponding condition as follows:

(51) (↑ TOPIC|FOCUS) = (↑ COMP* OBJ)

One can introduce a special symbol for TOPIC|FOCUS, which stands for a disjunction of discourse functions: DF. (51) can then be abbreviated as in (52):

(52) $(\uparrow \text{DF}) = (\uparrow \text{COMP}^* \text{ OBJ})$

The final version of the c-structure rule for fronting in English will therefore have the form of (53):[11]

(53)
$$\begin{array}{llll} \text{CP} & \rightarrow & \text{XP} & \text{C}' \\ & & (\uparrow \text{DF}) = \downarrow & \uparrow = \downarrow \\ & & (\uparrow \text{DF}) = (\uparrow \text{COMP}^* \text{ OBJ}) & \end{array}$$

In German, as well as objects, nearly any other constituent (e. g. subjects, sentential complements, adjuncts) can be fronted. The c-structure rule for this is shown in (54):[12]

(54)
$$\begin{array}{llll} \text{CP} & \rightarrow & \text{XP} & \text{C}' \\ & & (\uparrow \text{DF}) = \downarrow & \uparrow = \downarrow \\ & & (\uparrow \text{DF}) = (\uparrow \text{COMP}^* \text{ GF}) & \end{array}$$

Here, GF is an abbreviation for a disjunction of grammatical functions which can occur in the prefield.

7.6 Summary and classification

LFG is a constraint-based theory and utilizes feature descriptions and PSG rules. Grammatical functions are treated as primitives of the theory, which sets LFG apart from most of other theories covered in this book. They are not defined structurally (as in GB). LFG is a lexicalist theory. Like GPSG, LFG can do without transformations. Processes affecting argument structure such as passivization are analyzed by means of lexical rules. Whereas GPSG treated long-distance dependencies using the percolation of information in trees, LFG uses functional uncertainty: a part of the f-structure is identified with another f-structure that can be embedded to an arbitrary depth. Coherence and completeness ensure that the long-distance dependency can be correctly resolved, that is, it ensures that a fronted object is not assigned to an f-structure which already contains an object or one in which no object may occur.

While LFG does contain a phrase-structural component, this plays a significantly less important role compared to other models of grammar. There are rules in which all constituents are optional and it has even been proposed for some languages that there are

[11] Note that the two disjunctions that are abbreviated by the respective occurrences of DF are independent in principle. This is unwanted. We want to talk about either a topic or a focus not about a topic and a focus in the mother f-structure. So additional machinery is needed to ensure that both occurrences of DF refer to the same discourse function.

[12] Berman (1996) uses the symbol ZP for symbols in the prefield rather than XP in (54). She formulates various phrase structure rules for ZPs, which replace ZP with NP, PP, AP and various adjuncts. Following Berman, ZPs can also be combined with the verb in the middle field. For reasons of exposition, I refrained from using ZP symbols in the formulation of the VP rule (40) in Section 7.4 and instead used NP directly.

rules where the part of speech of the constituents is not specified (see Section 13.1.2). In these kinds of grammars, f-structure, coherence and completeness work together to ensure that the grammar only allows well-formed structures.

LFG differs from other theories such as HPSG and variants of Construction Grammar in that feature structures are untyped. Generalizations can therefore not be represented in type hierarchies. Until a few years ago, the hierarchical organization of knowledge in inheritance hierarchies did not form part of theoretical analyses. In computer implementations, there were macros but these were viewed as abbreviations without any theoretical status. It is possible to organize macros into hierarchies and macros were discussed explicitly in Dalrymple, Kaplan & King (2004) with reference to capturing linguistic generalizations. Asudeh, Dalrymple & Toivonen (2008) suggest using macros not only for the organization of lexical items but also for capturing generalizations regarding c-structure annotations. Because of these developments, there was a greater convergence between LFG and other theories such as HPSG and CxG.

Williams (1984) compares analyses in LFG with GB. He shows that many analyses are in fact transferable: the function that f-structure has in LFG is handled by the Theta Criterion and Case Theory in GB. LFG can explicitly differentiate between subjects and non-subjects. In GB, on the other hand, a clear distinction is made between external and internal arguments (see Williams 1984: Section 1.2). In some variants of GB, as well as in HPSG and CxG, the argument with subject properties (if there is one) is marked explicitly (Haider 1986a; Heinz & Matiasek 1994; Müller 2003b; Michaelis & Ruppenhofer 2001). This special argument is referred to as the *designated argument*. In infinitival constructions, subjects are often not expressed inside the infinitival phrase. Nevertheless, the unexpressed subject is usually coreferential with an argument of the matrix verb:

(55) a. Er versucht, [das Buch zu lesen].
he tries the book to read
'He is trying to read the book.'

b. Er zwingt ihn, [das Buch zu lesen].
he forces him the book to read
'He is forcing him to read the book.´

This is a fact that every theory needs to be able to capture, that is, every theory must be able to differentiate between subjects and non-subjects.

For a comparison of GB/Minimalism and LFG/HPSG, see Kuhn (2007).

Comprehension questions

1. What do the terms *coherence* and *completeness* mean?
2. What are extended head domains?
3. What does lexical integrity mean?

Exercises

1. Give the lexical entry for *kannte* 'knew'.

2. How could one analyze the following sentence?

   (56) Den Apfel verschlingt David.
   the apple devours David
   'David devours the apple.'

   Provide the necessary c-structure rules. What kind of f-structure is licensed? Draw a syntactic tree with corresponding references to the f-structure. For fronted constituents, simply write NP rather than expanding the XP node. The c-structure rule for the NP can also be omitted and a triangle can be drawn in the tree.

Further reading

Section 7.1 was based extensively on the textbook and introductory article of Dalrymple (2001, 2006). Additionally, I have drawn from teaching materials of Jonas Kuhn from 2007. Bresnan (2001) is a comprehensive textbook in English for the advanced reader. Some of the more in-depth analyses of German in LFG are Berman (1996, 2003a).

Levelt (1989) developed a model of language production based on LFG. Pinker (1984) – one of the best-known researchers on language acquisition – used LFG as the model for his theory of acquisition. For another theory on first and second language acquisition that uses LFG, see Pienemann (2005).

Schwarze & de Alencar (2015) demonstrate how the XLE system can be used for the development on a French LFG grammar. The textbook also discusses the Finite State Morphology component that comes with the XLE system.