

# Lexical Functional Grammar as a Construction Grammar

*Jamie Y. Findlay*  
University of Oslo

## ABSTRACT

Lexical Functional Grammar (LFG) is a lexicalist, constraint-based grammatical theory that shares a lot of the basic assumptions of Construction Grammar (CxG), such as a commitment to surface-oriented descriptions (no transformations), and the simultaneous representation of form, meaning, and other grammatical information (no derivations). Nevertheless, LFG is not standardly viewed as a kind of CxG, in particular since its adherence to the principle of Lexical Integrity means that it insists on a strict morphology-syntax distinction where CxG canonically rejects such a divide. However, such a distinction is in fact entirely compatible with CxG assumptions; the actual problem with viewing LFG as a CxG is the difficulty it has in describing the more substantive end of the schematic-substantive spectrum of constructions. I suggest that by replacing the limited context-free grammar base of LFG responsible for this shortcoming with a more expressive formalism (in this case a description-based tree-adjoining grammar), we can obtain a fully constructional LFG, suitable as a formal framework for CxG.

*Keywords: Lexical  
Functional  
Grammar,  
Construction  
Grammar, Lexical  
Integrity,  
meta-theory*

## INTRODUCTION

1

In grammatical theory, there is an important division between which parts of linguistic competence involve storage/memory and which involve computation. Exactly where the line between these two categories should be drawn is an open, and heavily debated, question.

The creativity and productivity of human language means that it is untenable to claim that everything is stored; if this were the case, it would be impossible to say anything new – we would only be able to repeat what we had already heard and memorised. This is the aspect of human language emphasised by work in mainstream generative grammar (MGG),<sup>1</sup> and claims about the ‘discrete infinity’ of human language are commonly seen in the opening pages of textbooks which introduce students to natural language syntax from this perspective. On the other hand, the arbitrariness in human language means that we cannot claim that everything is computed, either: some form-meaning pairings are the way they are for no other reason than convention, and conventions must be learned. This is the Saussurean observation about the arbitrariness of the linguistic sign: there is no more reason for *dog* to be used to refer to the domestic canine than any other sequence of sounds, which is precisely why languages vary in this respect (e.g. the German word for the same concept is *Hund*, the French *chien*, etc.). This arbitrariness also exists above the level of the word (see below), and it is this aspect of human language which is emphasised by work in the tradition of Construction Grammar (CxG).<sup>2</sup>

A traditional view in MGG is that the things which are stored are words, collected in the LEXICON, and that objects larger than the word – phrases, clauses, etc. – are arrived at by the application of general and abstract rules of syntactic composition to these lexical atoms. One problem with this view is that the arbitrariness of natural language does not stop at the word level: there are a variety of phrasal objects which do not behave as we would expect from the normal syntactic processes of the language in question, and whose meanings (and sometimes forms) therefore apparently have to be memorised. The most striking examples are idioms, whose meanings are often wholly

---

<sup>1</sup> This term is borrowed from Culicover and Jackendoff (2005), and is used to refer to work in the Chomskyan tradition, i.e. that strand of theoretical thinking that begins with *Syntactic structures* (Chomsky 1957) and continues to the present day with work in the Minimalist Program (Chomsky 1993, 1995).

<sup>2</sup> CxG refers to a family of theories which originated in the work of Charles Fillmore and colleagues in the 1980s (Fillmore 1985, 1988; Fillmore *et al.* 1988; Kay and Fillmore 1999), and that recognise the construction, a pairing of form and meaning of arbitrary size and varying abstraction, as the basic unit of analysis in grammatical theory. More details will be given about CxG below.

unpredictable given the meaning of their parts in other contexts (if they even exist outside the idiom), and whose syntactic structures are often anomalous or archaic:

- (1) Holden **pulled a rabbit out of the hat**.  
≈ ‘Holden found an unexpected solution to the problem.’  
(Anomalous semantics: no rabbits or hats need be involved.)
- (2) The thieves have **flown the coop**.  
≈ ‘The thieves have escaped.’  
(Anomalous semantics: no coops or flying need be involved.  
Anomalous syntax: *fly* does not normally take a Source direct object in contemporary English.)
- (3) We’ve tried **every which way** to solve this problem, and there’s just no solution.  
≈ ‘We’ve tried every possible means of solving this problem, and there’s just no solution.’  
(Anomalous syntax; not possible with other quantifiers, for example: \**each which way*.)
- (4) We’ve let these pirates **run amok** for too long.  
≈ ‘We’ve let these pirates cause chaos for too long.’  
(Anomalous lexical content: *amok* does not exist outside of this expression.)

But there are more schematic phrasal configurations which also bear unpredictable meanings, illustrated in (5)–(7):

- (5) The more you tighten your grip, Tarkin, the more star systems will slip through your fingers.  
≈ ‘As you tighten your grip, the number of star systems slipping through your fingers will correspondingly increase.’  
(The comparative correlative/*the X-er the Y-er* construction; Fillmore 1987; Culicover and Jackendoff 1999.)
- (6) What’s a nice girl like you doing in a place like this?  
≈ ‘How come a nice girl like you is in a place like this?’  
(The *what’s X doing Y/WXDY* construction; Kay and Fillmore 1999.)
- (7) Bill belched his way out of the restaurant.  
≈ ‘Bill left the restaurant while belching.’  
(The *way*-construction; Jackendoff 1992.)

Fillmore *et al.* (1988, 505–506) call these FORMAL IDIOMS, in contrast to the SUBSTANTIVE IDIOMS in (1)–(4). Formal idioms have more open slots which can be filled in with any appropriate word or phrase, whereas substantive idioms require specific words for their idiomatic meaning to come off. In fact, idioms exist on a spectrum from more substantive to more formal (or schematic).

Because of the existence of these larger-than-single-word expressions whose meaning and/or form cannot be computed on-line, CxG takes a different view from MGG: the building blocks of phrasal syntax are not words, but CONSTRUCTIONS, pairings of form and meaning of *any* size. Word-internal syntax is also often understood to fall under this umbrella, so that constructions extend both above and below the level of the word, with the distinction between phrasal syntax and the lexicon therefore breaking down. Instead, the grammar is simply a repository of constructions – the CONSTRUCTICON (Jurafsky 1992) – and some means of combining them (often unification, since constructions are often represented as feature structures).

There is quite some diversity in how this insight is cashed out, leading to an array of sometimes quite disparate theories all bearing the moniker ‘Construction Grammar’, e.g. Berkeley Construction Grammar (Fillmore 1985, 1988; Kay and Fillmore 1999), Embodied Construction Grammar (Bergen and Chang 2005), Fluid Construction Grammar (Steels 2011; Steels and van Trijp 2011), Sign-Based Construction Grammar (Sag 2010; Boas and Sag 2012; Michaelis 2015), Cognitive Construction Grammar (Lakoff 1987; Goldberg 1995, 2006), and Radical Construction Grammar (Croft 2001). Nonetheless, there are certain overriding meta-theoretical assumptions that basically all CxGs have in common, which are identified below:

1. WYSIWYG: Linguistic descriptions are surface oriented, or ‘WYSIWYG’ (‘What You See Is What You Get’) in nature – that is, no phonologically empty elements are assumed, and there is no abstract ‘underlying’ form which must be transformed to reach the surface representation.
2. PARALLEL-REPRESENTATION: All levels of linguistic analysis, both in terms of form and meaning, are present in parallel – that is, no level of representation is derived from another (e.g. meaning is not derived from form, nor *vice versa*).

3. EDL: Linguistic description has an ‘extended domain of locality’ – that is, the notion of the Saussurean sign extends above the level of the word, and form-meaning pairings can exist which necessitate dependencies between structurally distant parts of a sentence.
4. CONSTRUCTIONS-ALL-THE-WAY-DOWN: Since the notion of sign also extends *below* the level of the word, a corollary of EDL is that there is no absolute/discrete distinction between morphology and syntax, since words and phrases are built out of the same things: “it’s constructions all the way down” (Goldberg 2006, 18).
5. HIERARCHY: Linguistic knowledge is structured, and organised in a hierarchical fashion – often in some kind of inheritance network or type hierarchy, of the sort also assumed to structure non-linguistic knowledge.
6. CROSS-LX-VARIETY: There is a greater emphasis on cross-linguistic variety, on ‘unusual’ constructions, and on subtle connections of form and meaning than is found in MGG, for instance, where the focus is much more on ‘core’ constructions and cross-linguistic similarity.
7. USAGE-BASED: Knowledge of language is based on usage – that is, there is no strict competence-performance distinction, and we store both linguistic generalisations and specific episodic memories of linguistic events.

Within these assumptions, we can draw a dividing line between the first five, which are more about the architecture of the grammar, and the final two, which are about what you do with that grammar – i.e. what kinds of questions linguists should be asking, and where they should look for their explanations.

In this paper, I want to argue that Lexical Functional Grammar (LFG: Kaplan and Bresnan 1982; Bresnan *et al.* 2016; Dalrymple *et al.* 2019), a constraint-based, declarative grammatical theory, can be seen as another kind of Construction Grammar – or, more precisely, that it can be viewed as a suitable framework for formalising CxG ideas and analyses.<sup>3</sup> For this reason, I will be focussing on the first five

---

<sup>3</sup> Cf. Lichte and Kallmeyer (2017) and Müller (2021), who undertake a similar exercise for Tree-Adjoining Grammar (TAG) and Head-Driven Phrase Structure Grammar (HPSG), respectively.

assumptions above, since my interest is in the formal properties of LFG as a system, rather than to what ends researchers make use of the framework.<sup>4</sup>

I begin in Section 2 with an introduction to LFG, highlighting its key features and pointing out to what extent these allow it to satisfy assumptions 1–5 above. It will be seen that it already satisfies all of them to some extent, with the notable exception of CONSTRUCTIONS-ALL-THE-WAY-DOWN: LFG self-avowedly adheres to the principle of LEXICAL INTEGRITY (LI), which means that it rejects the claim that there is no distinction between morphology and syntax.

In Section 3, however, I argue that *some* version of LI should be adopted by CxG (and already is in frameworks like SBCG), and therefore that the assumption of CONSTRUCTIONS-ALL-THE-WAY-DOWN ought not to be a *sine qua non* of CxG. On the other hand, I also argue that LFG should (and sometimes already does, albeit often implicitly) loosen the absolute distinction between morphology and syntax, since some apparent LI violations do seem to be genuine.

In fact, the problem with viewing LFG as a formalisation of CxG lies not with LI, but with its inability to handle substantive idioms satisfactorily, owing to the difficulty of describing multiword stretches in the lexicon. Section 4 examines how LFG handles some constructional phenomena, showing that existing machinery allows it to analyse many formal idioms well, but that it falls short when it comes to substantive idioms. I discuss some existing inadequate proposals, and conclude that Findlay's (2019; to appear) proposal to replace the context-free grammar backbone of LFG with a tree-adjoining grammar would give the appropriate level of descriptive freedom to enable LFG to capture substantive idioms. With this move, LFG's notion of 'extended domain of locality' is expanded to include phrase struc-

---

<sup>4</sup>In its guise as a research paradigm rather than a formalism, LFG has tended to be split on assumptions 6 and 7. Cross-linguistic variety has been a major focus, especially of the Parallel Grammar project (ParGram; Butt *et al.* 2002), and non-configurationality has provided an important motivation for LFG's modular architecture (see e.g. Bresnan *et al.* 2016, ch. 1). In keeping with its generative roots, however, LFG researchers tend to treat the competence/performance distinction as a given – although see work in LFG-DOP (Bod and Kaplan 1998) for a more usage-based approach.

ture, meaning that lexical entries become nothing more or less than a declarative description of every level of linguistic structure in parallel – exactly what we would expect from a Construction Grammar.

## LEXICAL FUNCTIONAL GRAMMAR

2

LFG is a declarative/constraint-based (i.e. non-transformational) grammatical theory, an off-shoot of MGG stemming from a desire in the late 1970s and early 1980s to develop a more psychologically plausible and computationally tractable theory (Kaplan and Bresnan 1982, 173–174; Dalrymple and Findlay 2019, 123).<sup>5</sup> In this section, I introduce some of its key machinery while considering how well it adheres to the assumptions of CxG identified in Section 1.<sup>6</sup> We will see that LFG in its canonical form already shares many of them. Assumption 5, HIERARCHY, is not met by LFG ‘out of the box’, but is easily accommodated with the addition of TEMPLATES, a tool already common in computational work in LFG, and now gaining ground in theoretical work (to be introduced in Section 2.3). The status of EDL and its supposed corollary CONSTRUCTIONS-ALL-THE-WAY-DOWN is more challenging: we will see in Section 2.1 that LFG has very powerful tools for describing non-local relationships; however, LFG’s adherence to Lexical Integrity means that it assumes a strict and categorical distinction between lexicon and

---

<sup>5</sup> Although the focus on psychology has not been sustained in contemporary work, LFG does still play a role in psycholinguistic work – for instance, some of its insights underpin the influential Bock-Levelt model of language production (Bock and Levelt 1994). There has been a far bigger focus on computational implementation, most notably through the Xerox Linguistic Environment (XLE: Kaplan and Newman 1997; Crouch *et al.* 2017), in which grammars for a very large number of different languages have been written as part of the Parallel Grammar (ParGram) project (Butt *et al.* 2002).

<sup>6</sup> Of course, this will by necessity be a fairly superficial introduction. For further details, the reader is directed to the relevant parts of Dalrymple *et al.* 2019, and to Dalrymple forthcoming. For an article-length overview of LFG, see Asudeh and Toivonen 2015, and for textbook-style introductions, see Bresnan *et al.* 2016 and Börjars *et al.* 2019.

grammar, contrary to CONSTRUCTIONS-ALL-THE-WAY-DOWN (Section 3), and the format of LFG's lexical entries hampers its commitment to EDL by making it impossible to handle certain kinds of constructions, especially substantive idioms, in a satisfactory way (Section 4.3).

## 2.1 *Two levels of syntactic structure*

What is generally called 'syntax' refers both to more imminent, 'surfacey' phenomena such as word order (which vary widely across the world's languages) and to more abstract, 'deeper' phenomena such as subjecthood (which exhibit many more commonalities cross-linguistically). LFG formalises this distinction by positing two distinct levels of syntactic structure, C(ONSTITUENT)-STRUCTURE and F(UNCTIONAL)-STRUCTURE, which encode the different kinds of information in different data structures, and which are related by correspondence (not by derivation). C-structure is a phrase-structure tree, and represents constituency, part-of-speech categories, and word order. F-structure is a feature structure/attribute-value matrix (AVM), and represents abstract relational information about grammatical functions, agreement, long-distance dependencies, etc. The two are connected by a PROJECTION FUNCTION,  $\phi$ , which maps c-structure nodes onto their corresponding f-structure ( $\phi$  is a function, so more than one c-structure node can be mapped to the same f-structure, but each c-structure node only maps to a single f-structure). Figure 1 gives the c- and f-structures for the sentence *Jadzia loves them* by way of illustration.<sup>7</sup>

C-structure is loosely based on X-bar theory (Chomsky 1970; Jackendoff 1977), but makes a number of simplifying assumptions:<sup>8</sup>

---

<sup>7</sup> Figure 1 only represents the  $\phi$  function on the three c-structure nodes which correspond to maximal functional projections (and which map to the three f-structures), so as to avoid visual clutter (see Dalrymple and Findlay 2019, 137–138). This does not conceal any information, since daughter nodes in each of these three projections will be annotated to indicate that they share the same f-structure as their mother, with the effect that their functional information is 'passed up' the tree – see below for explanation of annotations on c-structure.

<sup>8</sup> For a fuller account of the formal details of c-structure, see Dalrymple *et al.* 2019, ch. 3.

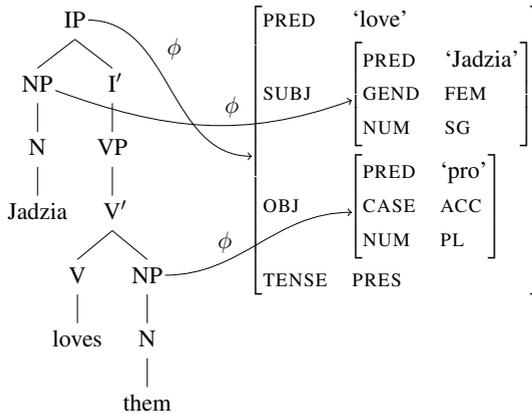


Figure 1:  
C- and f-structures  
for *Jadzia loves them*,  
with correspondences

1. All right-hand elements of phrase-structure rules are optional, so that there can be headless phrases (VPs without Vs, IPs without Is, etc.) – this avoids positing empty heads where no overt material fills the slot (e.g. in the analysis of English, an I node is only assumed when there is an auxiliary or modal that fills it, otherwise it is simply omitted).
2. Some categories are NON-PROJECTING (Toivonen 2003), indicated by a circumflex accent over the category label: e.g. a  $\hat{P}$  is a non-projecting P. This means that they do not project a phrase.
3. We assume there is always a rule  $XP \rightarrow X$ , for any category X, which omits extraneous bar levels (this is part of a general principle called ECONOMY OF EXPRESSION; see Dalrymple *et al.* 2015).

In addition, no phonologically empty elements are assumed.<sup>9</sup> All of this makes LFG c-structures a much more direct representation of

<sup>9</sup>Some versions of LFG violate this by employing traces. This was common in earlier incarnations of the theory, including Kaplan and Bresnan 1982, along with e.g. Bresnan 1995, 1998; but since the introduction of functional uncertainty (Kaplan *et al.* 1987; see below), it is not normally seen as part of the mainstream theory. Various handbook and textbook presentations of LFG, such as Dalrymple 2001, Börjars *et al.* 2019, and Dalrymple *et al.* 2019, do not employ traces, for example. Awkwardly, one prominent textbook, Bresnan *et al.* 2016, *does* make use of empty categories, albeit only in a heavily restricted set of cases (such as crossover phenomena – see Bresnan *et al.* 2016, ch. 9). However, others have convincingly argued for alternative analyses of these phenomena which remove the need for traces and empty categories in LFG altogether: see Dalrymple *et al.* 2001, 2007, Dalrymple and King 2013, and Nadathur 2013.

surface syntactic structure than trees in other frameworks where the phrase-structure tree is overloaded and expected to encode functional information as well as constituent structure. LFG takes the view that different kinds of information require different kinds of data structure to represent: syntactic trees are very good at encoding constituency and linear order, but much less good at representing dependency relations between constituents (which may involve re-entrancy, cyclicity, etc.), for which a feature structure is much better suited. By omitting abstract functional information from the tree, we therefore obtain a much more WYSIWYG c-structure: assumption 1 of CxG described above.<sup>10</sup>

Formally, an LFG grammar is a context-free grammar where the phrase-structure rules bear annotations that describe how f-structure is projected from the c-structure. Annotations are written using the following abbreviations:

- (8) a.  $*$  := the current node (the node bearing the annotation)  
 b.  $\hat{*}$  := the current node's mother
- (9) a.  $\downarrow$  :=  $\phi(*)$  (the f-structure of the current node)  
 b.  $\uparrow$  :=  $\phi(\hat{*})$  (the f-structure of the current node's mother)

We can indicate that a node and its mother share the same f-structure by writing  $\uparrow = \downarrow$ :

$$(10) \quad \text{NP} \rightarrow \quad \text{N} \\ \uparrow = \downarrow$$

And we can indicate that a phrase bears some particular grammatical function by using paths through f-structure in our annotations. The rule in (11), for example, says that the f-structure of the NP in the specifier of IP is the SUBJECT of the f-structure corresponding to the IP:<sup>11</sup>

$$(11) \quad \text{IP} \rightarrow \quad \text{NP} \quad \text{I}' \\ (\uparrow \text{SUBJ}) = \downarrow \quad \uparrow = \downarrow$$

<sup>10</sup>Zaenen (1989) makes this explicit in a 'WYSIWYG Principle'.

<sup>11</sup>For reasons of space, I will not motivate or list the grammatical functions and features usually assumed at f-structure. For a full treatment, see Dalrymple *et al.* 2019, ch. 2.

In words, the annotation below the NP says that the f-structure corresponding to its mother node, IP (“↑”), has a subject attribute (“SUBJ”), whose value is the NP’s f-structure (“↓”).

It is important to recognise that although we say that f-structure is *projected* from c-structure, this does not mean that f-structure is *derived* from c-structure in any sense. Rather, the two structures are both simultaneously present, and constrain each other *mutually*. (This is an important component in LFG’s adherence to PARALLEL-REPRESENTATION, which we return to in Section 2.2, where I introduce the wider LFG projection architecture.) The directionality inherent in the projection function is related to information flow rather than derivation: owing to the functional nature of  $\phi$ , structure present at c-structure can be lost at f-structure (many nodes can correspond to a single f-structure), and cannot then be recovered in reverse (in the same way that mergers are irreversible in sound change).

Lexical entries in LFG are formally just phrase-structure rules, as in (12):

- (12)    N →            Jadzia  
           (↑ PRED) = ‘Jadzia’  
           (↑ NUM) = SG  
           (↑ GEND) = FEM

But they are usually written in a different format, shown in (13):

- (13)    Jadzia    N    (↑ PRED) = ‘Jadzia’  
                           (↑ NUM) = SG  
                           (↑ GEND) = FEM

The feature PRED was originally used to indicate the semantic predicate of an f-structure, but given developments in the LFG approach to semantics (see Section 2.4), its role is now really just to uniquely identify lexical items (see Andrews 2008 and Findlay 2019, 152–154 for some discussion) – cf. the role of the LID feature in SBCG (Sag 2012, 84). Lexical entries therefore almost always contain a statement identifying their PRED value at a minimum.

Annotations, whether in lexical entries or other phrase-structure rules, can refer to non-local parts of f-structure. We have seen how annotations can include paths through f-structure; in principle there is no



very precise specifications of very detailed and complex long-distance relationships through the f-structure. Thus, the description language of LFG allows constraints to be placed on dependencies that extend way beyond a word and its immediate sisters or dependents, which clearly affords LFG some version of EDL, assumption 3 of CxG described above.

*The parallel projection architecture*

2.2

Although the original formulation of LFG in Kaplan and Bresnan 1982 includes only c-structure and f-structure, subsequent developments have expanded the number of different levels of representation, i.e. the different ‘structures’, which are assumed. A contemporary view of the so-called (PARALLEL) PROJECTION ARCHITECTURE is given in Figure 2, showing the different structures and correspondence functions which map between them. All of these different structures are taken to have “their own primitives and organizing principles, and therefore

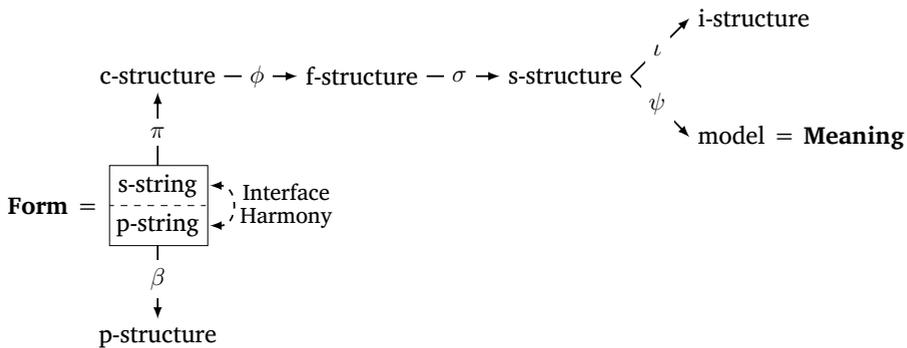


Figure 2: The parallel projection architecture (Findlay 2021, 344). On the division of the string into the s-string and p-string, see Dalrymple and Mycock 2011 and Mycock and Lowe 2013. The other structures shown here are p(rosodic)-structure (Mycock and Lowe 2013), s(emantic)-structure (Dalrymple 1999; Lowe 2014; Findlay 2021), and i(nformation)-structure (Dalrymple and Nikolaeva 2011). Not shown are a(rgument)-structure, which appears between c-structure and f-structure in some conceptions of the architecture (Butt *et al.* 1997), but which other approaches have omitted entirely (Asudeh and Giorgolo 2012; Findlay 2016); and m(orphyntactic)-structure (Butt *et al.* 1996; Frank and Zaenen 2004), which has likewise been dispensed with in modern treatments (Dalrymple 2015)

their own internal structure and formal representation” (Dalrymple *et al.* 2019, 265 – although in practice almost all are represented as AVMs like f-structure), meaning that LFG takes a highly modular view of the grammar. Crucially, meaning is also included in this extended view of the LFG architecture, as well as information-structure, so that all kinds of form *and* function are, at least in principle, brought within the scope of the framework. This shows that LFG shares assumption 2 of CxG, PARALLEL-REPRESENTATION: all levels of linguistic analysis, both form and function, are represented simultaneously.

What is more, although each of these structures represents a different plane of linguistic analysis, they are not derived from one another; instead they are present in parallel, and are mutually constraining. Just as phrase-structure rules can be annotated to describe f-structure, they can also bear annotations referring to any level, e.g. a person’s name like *Jadzia* might specify that the ANIMATE feature in its s-structure has the feature +:

$$(16) \quad \text{Jadzia} \quad \text{N} \quad (\uparrow \text{ PRED}) = \text{‘Jadzia’}$$

$$(\uparrow_{\sigma} \text{ ANIM}) = +$$

The subscript convention here is used to make such annotations more readable.  $\uparrow_{\sigma}$  is equivalent to  $\sigma(\uparrow)$ , and, more generally, for any structure  $s$  and any projection function  $\omega$ ,  $s_{\omega} := \omega(s)$ . Such subscripts can also be iterated, so that, for instance  $\downarrow_{\sigma^i}$  is equivalent to  $\iota(\sigma(\downarrow))$ , or, in other words, this node’s i-structure.

Thus, descriptions (on both phrase-structure rules and in lexical entries) can constrain all levels of representation simultaneously – or, rather, all levels except c-structure. For, since the annotations appear on phrase-structure rules in a context-free grammar, the scope of c-structure constraints remains within a single generation (a mother node and its daughters). We will return to this problem in Section 4.3.

### 2.3

### *Templates*

It is common in programming languages to use *macros* of some kind to abbreviate chunks of code when they will be repeated. This has the pragmatic benefit of saving typing time, but it also makes maintaining code much easier: if something has to be changed in the chunk of code

in question, it need only be changed in one place, where the macro is defined, rather than having to be changed at every instance of its use. This saves time too, but, more importantly, it also avoids errors being introduced when some instances are inevitably missed.

The computational implementation of LFG, the Xerox Linguistic Environment (XLE: Kaplan and Newman 1997; Crouch *et al.* 2017), also provides a means of writing macros – in this case they are called TEMPLATES. In addition to their practical uses, there has, over the past two decades, been a growing interest in the *theoretical* applications of templates in LFG, as a means of expressing generalisations across different lexical entries or parts of the grammar (Dalrymple *et al.* 2004; Asudeh *et al.* 2014; Findlay 2020, 132–133). Since templates are just abbreviations, a grammar with templates is extensionally equivalent to one without, but the former will be able to express generalisations that the latter cannot.

One area where templates can capture generalisations is in abbreviating annotations that frequently co-occur. For instance, any distinctively third-person singular verb in English will share the second and third lines of this lexical entry for *loves*:<sup>12</sup>

- (17) loves V (↑ PRED) = ‘love’  
 (↑ SUBJ PERS) = 3  
 (↑ SUBJ NUM) = SG

We can therefore define a template 3SG-SUBJECT that abbreviates this information:

- (18) 3SG-SUBJECT :=  
 (↑ SUBJ PERS) = 3  
 (↑ SUBJ NUM) = SG

Now we can rewrite the lexical entry for *loves* by ‘calling’ this template, indicated by prefixing the template name with an ‘@’ symbol:

---

<sup>12</sup>These annotations illustrate how agreement works in LFG: an agreeing subject (e.g. *Jadzia loves ...*) will provide the same values for these features as the verb does, meaning the specifications are compatible; by contrast, a non-agreeing subject (e.g. *\*We loves ...*) will cause a feature clash in its f-structure, since it will specify different values for its PERSON and NUMBER features (e.g. 1 and PL in this case).

- (19) loves V ( $\uparrow$  PRED) = ‘love’  
@3SG-SUBJECT

Templates can be parametrised, as in (20), so that they take one or more arguments, allowing even more flexibility:

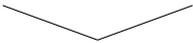
- (20) TENSE( $t$ ) :=  
( $\uparrow$  TENSE) =  $t$

Notably, templates can also be nested, as shown in (21) and (22):

- (21) a. 3-SUBJECT :=  
( $\uparrow$  SUBJ PERS) = 3  
b. SG-SUBJECT :=  
( $\uparrow$  SUBJ NUM) = SG

- (22) 3SG-SUBJECT :=  
@3-SUBJECT  
@SG-SUBJECT

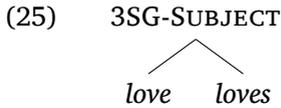
That is, a template can call one or several other templates in its definition. This nesting creates an implicit hierarchy between templates:

- (23) 3-SUBJECT SG-SUBJECT  
  
3SG-SUBJECT

Such a hierarchy is different from a typical inheritance hierarchy – it is an *inclusion* hierarchy instead (Asudeh *et al.* 2013, 17–19). This is because templates, in common with LFG descriptions generally, allow the use of Boolean operators like negation or disjunction. For example, alongside the 3SG present tense form *loves*, we have the complementary form *love*, used for all other person/number combinations in the present tense. We can describe this distribution by simply negating the 3SG-SUBJECT template:

- (24) love V ( $\uparrow$  PRED) = ‘love’  
 $\neg$ @3SG-SUBJECT

But now both *love* and *loves* will be daughters of 3SG-SUBJECT in the template hierarchy, since both *include* the template, even though in one case this is only under negation:



In principle, all functional annotations in a grammar could be abbreviated in templates and appropriately related to one another. This would provide LFG with a hierarchical organisation of linguistic knowledge, bringing it in line with assumption 5 of CxG, HIERARCHY.<sup>13</sup> In practice, however, theoretical work in LFG has not pursued this endeavour in a thoroughgoing way (though see Asudeh *et al.* 2013 and Przepiórkowski 2017 for case studies), and so the framework does not yet live up to the claim by Goldberg that in CxG “the network of constructions captures our grammatical knowledge of language *in toto*” (Goldberg 2006, 18) – although only for contingent rather than principled reasons.

#### Meaning in LFG

2.4

Any theory which purports to explain human language needs to have an account of meaning. In particular, it needs to explain how meanings can be arrived at compositionally, allowing us to express new ideas with existing, limited, resources. LFG remains wholly agnostic about how meanings *per se* should be represented – in keeping with its modular approach, this is not a question for the framework overall, but for the particular module which deals with semantics. What is crucial, though, is how this module connects to the rest of the grammar: in other words, the syntax-semantics interface. There has been some variation over the years in how this has been conceptualised within LFG, and in particular about the necessity and/or role of s-structure in this (on which see Findlay 2021, especially §3), but the *de facto* standard approach to the syntax-semantics interface in contemporary LFG is GLUE SEMANTICS (Glue: Dalrymple *et al.* 1993; Dalrymple 1999).

<sup>13</sup>Work in CxG generally makes use of *inheritance* hierarchies, and therefore LFG’s template inclusion hierarchies may not seem like such a good fit. It remains an open question, however, what exactly the hierarchical structure of the grammar should look like, and further work is needed to determine the theoretical implications of choosing an inclusion rather than an inheritance hierarchy.

For our purposes, most of the details of this theory are not relevant, but it will nonetheless be useful to have some tools to describe how LFG handles the pairing of form and meaning, and so in this section I give a brief introduction to Glue for LFG. For a fuller introduction to the theory, the reader is directed to Asudeh 2012, ch. 4 or Dalrymple *et al.* 2019, ch. 8.

Meaning contributions in Glue are handled by so-called MEANING CONSTRUCTORS, which pair an expression in some meaning language (here a simple predicate calculus) with a logical expression that both gives the type of that meaning and connects it to the syntax – this logical expression is called the glue term, since it bonds the semantics to the syntax. Semantic composition is logical deduction: parsing a sentence gives us a collection of meaning constructors, and we use their glue terms to construct a proof terminating in the type of the sentence itself.

Glue Semantics uses LINEAR LOGIC (Girard 1987) as the logical language for the second part of a meaning constructor. Since it lacks the structural rules of weakening and contraction, this logic has the property of RESOURCE SENSITIVITY, meaning that premises are ‘used up’ in deriving a conclusion. This has the – desirable – consequence that meanings cannot be re-used or discarded in the process of composition. For example, *Jadzia loves Worf* cannot mean  $\text{love}(\text{jadzia}, \text{jadzia})$  (‘Jadzia loves herself’), where we use the meaning of *Jadzia* twice and ignore the meaning of *Worf*.<sup>14</sup>

A simple meaning constructor is given in (26):

(26)  $\text{jadzia} : e_{\uparrow}$

The meaning language side introduces a constant *jadzia*, while the linear logic side says that this is of type  $e$  and is associated with  $\uparrow$ : in a lexical entry this means the pre-terminal node’s f-structure, i.e. the lexical item’s own f-structure.<sup>15</sup>

---

<sup>14</sup>On the more widespread relevance of resource sensitivity to linguistic theory, see Asudeh 2012, ch. 5.

<sup>15</sup>In much of the Glue Semantics literature, types are associated with s-structures rather than f-structures, but for our purposes f-structures are sufficient, and avoid us being drawn into unresolved disputes about the exact content

A more complex meaning constructor appears in (27):

$$(27) \quad \lambda x. \lambda y. \text{love}(x, y) : e_{(\uparrow_{\text{SUBJ}})} \multimap e_{(\uparrow_{\text{OBJ}})} \multimap t_{\uparrow}$$

This has a two-argument function on the left-hand side, and on the right-hand side a linear logic expression with two implications. This second part shows the  $\langle e, \langle e, t \rangle \rangle$  type of the function on the left ( $\multimap$  is right-associative), and also links its first argument to its f-structure subject and its second to its f-structure object. Read as an implication, the glue term can be thought of as saying the following: “If I am provided with something of type  $e$  corresponding to my subject, and if I am then provided with something of type  $e$  corresponding to my object, I will provide something of type  $t$  corresponding to my own f-structure (i.e. the f-structure of the clause)”.

Of course, combining types means nothing if we don’t also combine meanings. Glue achieves this by appealing to the CURRY-HOWARD CORRESPONDENCE (Curry and Feys 1958; Howard 1980): proof steps in a constructive logic (like linear logic) correspond to specific operations in the lambda calculus. Most notably, implication elimination (i.e. *modus ponens*) corresponds to functional application, while implication introduction (i.e. hypothetical reasoning) corresponds to lambda abstraction. This means that as we compose the types on the right-hand side of a meaning constructor, the left-hand meanings are also combined appropriately. Let us see how this works with an example.

Meaning constructors are included in lexical entries just like other annotations. For the sentence *Jadzia loves Worf*, we can assume the (very simplified) lexical entries in (28)–(30):

$$(28) \quad \text{Jadzia} \quad \text{N} \quad (\uparrow \text{PRED}) = \text{‘Jadzia’}$$

$$\text{jadzia} : e_{\uparrow}$$

---

of s-structure (on which see Findlay 2021, §3). I also make use of so-called FIRST-ORDER GLUE here (Kokkonidis 2008), where the atomic expressions in the linear logic are type constructors that take structures in the projection architecture (here f-structures) as arguments (here represented as subscripts), rather than the more common approach which takes the atoms to be the structures themselves (appropriately typed). This is mostly for the sake of clarity, since it makes the role of the linear logic in driving composition based on types more explicit (see also discussion in Kokkonidis 2008 and Findlay 2019, 181–183).

(29) Worf N ( $\uparrow$  PRED) = ‘Worf’  
**worf** :  $e_{\uparrow}$

(30) loves V ( $\uparrow$  PRED) = ‘love’  
 $\lambda x.\lambda y.\mathbf{love}(x, y) : e_{(\uparrow\text{SUBJ})} \multimap e_{(\uparrow\text{OBJ})} \multimap t_{\uparrow}$

The (abbreviated) f-structure for the sentence is shown in (31). The different structures have been labelled to allow us to refer to them directly.

(31) 
$$l \left[ \begin{array}{ll} \text{PRED} & \text{‘love’} \\ \text{SUBJ} & j \left[ \begin{array}{ll} \text{PRED} & \text{‘Jadzia’} \end{array} \right] \\ \text{OBJ} & w \left[ \begin{array}{ll} \text{PRED} & \text{‘Worf’} \end{array} \right] \end{array} \right]$$

We can now obtain a set of INSTANTIATED meaning constructors from the lexically contributed meaning constructors in (28)–(30) by replacing the descriptions on the linear logic side with the names of the f-structures they describe in (31):

(32) **jadzia** :  $e_j$   
**worf** :  $e_w$   
 $\lambda x.\lambda y.\mathbf{love}(x, y) : e_j \multimap e_w \multimap t_l$

Finally, we can use these to construct the proof in Figure 3, where each step corresponds to an instance of *modus ponens*/function application. As we can see, we arrive at the correct meaning for the sentence, namely **love(jadzia, worf)**.

Figure 3:  
 Glue proof for  
*Jadzia loves Worf*

$$\frac{\frac{\mathbf{jadzia} : e_j \quad \lambda x.\lambda y.\mathbf{love}(x, y) : e_j \multimap e_w \multimap t_l}{\lambda y.\mathbf{love}(\mathbf{jadzia}, y) : e_w \multimap t_l} \quad \mathbf{worf} : e_w}{\mathbf{love}(\mathbf{jadzia}, \mathbf{worf}) : t_l}$$

## 2.5

### Summary

We have now seen briefly some key components of LFG, and I believe this has illustrated how many of the core assumptions of CxG it already shares. Its surface-oriented syntax, represented at c-structure, means

that it adheres to WYSIWYG. The parallel projection architecture gives us PARALLEL-REPRESENTATION: all levels of linguistic structure exist in parallel, mutually constraining one another. The existence of tools like functional uncertainty gives us an important degree of EDL; although c-structure remains problematic, something we will explore more fully in Section 4, it is clear that overall LFG is perfectly capable of expressing a variety of complex constraints across arbitrary distances. Lastly, HIERARCHY can be achieved through the use of templates to organise and modularise linguistic description, even though this approach has not been followed through to completion in theoretical work in LFG.

One problem arises, however, when it comes to CONSTRUCTIONS-ALL-THE-WAY-DOWN. LFG professes to adhere to the principle of Lexical Integrity (LI), whereby syntax and morphology are strictly separated, and the morphological structure of words is invisible to the syntax. Crucially, the LI claim that “words are built out of different structural elements and by different principles of composition than syntactic phrases” (Bresnan and Mchombo 1995, 181) would appear to be at odds with the CxG credo that “no strict division is assumed between the lexicon and syntax” (Goldberg 1995, 7). Since “LFG assumes a strict version of the Lexical Integrity Principle” (Dalrymple *et al.* 2019, §3.3), this would seem to be a serious obstacle to viewing LFG as a CxG. In the next section, we will examine LI and see that it may not prove as great an obstacle as appearances would suggest.

## LEXICAL INTEGRITY

3

The principle of Lexical Integrity claims that the smallest items the (phrasal) syntactic component of the grammar can ‘see’ are words. That is, word-internal morphological structure is not accessible to the syntax, and so there is an important division between the syntax on the one hand and the lexicon on the other, which may also be taken as an important division between the computational system underlying syntax and that underlying morphology.

Building on the proposals of Chomsky (1970), the principle of Lexical Integrity was first formulated by Lapointe (1980, 8) as the Generalized Lexicalist Hypothesis:

(33) **Generalized Lexicalist Hypothesis:**

No syntactic rule may refer to elements of morphological structure.

In the forty plus years since this original formulation, many different versions have been proposed, but all make the same basic claim: there is some kind of ‘firewall’ between syntax and the lexicon, with the latter feeding the former, but not *vice versa*. Perhaps the most succinct specification of this is given by Anderson (1992, 84):

(34) **Principle of Lexical Integrity:**

The syntax neither manipulates nor has access to the internal structure of words.

A whole paradigm of linguistic theories exist, called LEXICALIST theories, which are defined by their adherence to the principle of Lexical Integrity – LFG is one such theory. LI has featured explicitly in LFG analyses from the very start (Bresnan 1982; Simpson 1983), and appears in textbook/handbook presentations of the theory (Falk 2001, 26; Bresnan *et al.* 2016, 92; Börjars *et al.* 2019, 28; Dalrymple *et al.* 2019, 135–136). And there are good *prima facie* reasons to believe that LI is valid: many phenomena that it predicts to be impossible are indeed so. For example, gapping can be applied to words but not sub-lexical elements (examples from Simpson 1991, 51):

- (35) a. John liked the play, and Mary, the movie.  
(gapping of *liked* permitted)  
b. \*John liked the play, and Mary dis- it.  
(gapping of *-liked* not permitted)

And sub-parts of words cannot be modified independently of the whole (examples from Williams 2007, 354):

- (36) a. How complete are your results?  
b. \*[How complete]-ness do you admire?

Although *how* can modify *complete* in (36a), it cannot do so when *complete* is part of a larger word, *completeness*, as in (36b). Note that the deviancy of (36b) is not because its meaning is incoherent: its meaning is perfectly grammatically expressed by (37a). And it is entirely possible for *how* to modify *complete* inside a nominal expression, provided that nominal expression is phrasal, as shown in (37b) (Williams 2007, 354):

- (37) a. What degree of completeness do you admire?  
b. How complete a record do you admire?

These data notwithstanding, CxG is often understood as rejecting a strict separation of morphology and syntax – this is the assumption I called CONSTRUCTIONS-ALL-THE-WAY-DOWN above. Since constructions are pairings of form and meaning, and morphemes also fit this description, there is therefore no fundamental distinction between morphemes and words. Rather, all constructions exist on a lexicon-syntax spectrum, varying in particular in terms of SCHEMATICITY, i.e. how much the phonological form is specified by the construction. At the more lexical end of the spectrum, we have words and morphemes, which are fully specified for phonological form (e.g. *cat* has the form /kæt/, at least in British English); at the more syntactic end, we have abstract phrasal constructions, which are radically underspecified for phonological form (e.g. the so-called N-P-N construction, exemplified in phrases like *hour by hour*, *cheek to cheek*, and *attack after attack*, which has a non-compositional semantics and imposes various restrictions on its parts – the nouns must be count nouns, cannot have a determiner, etc. – but is compatible with a wide variety of nouns and (a more limited variety of) prepositions: see Jackedoff 2008).

Taken naïvely, therefore, the LFG and CxG positions are clearly incompatible, and so LFG would be unsuitable as a formalisation of CxG. But whether or not LI is valid is an empirical question, not (just) a matter of formalism. And answering it would resolve the rift between LFG and CxG one way or the other. If it is valid, then CxG should abandon CONSTRUCTIONS-ALL-THE-WAY-DOWN in its strictest interpretation and move closer to LFG. Alternatively, if it is not, then LFG should abandon LI and move closer to CxG. So, what are the facts?

Alongside the putative evidence in favour of LI presented above, there is also apparently equally clear counterevidence. For instance, phrases and even entire clauses can host derivational and inflectional suffixes in English:

- (38) a. His general [ok-with-less-than-we-should-aim-for]-ness makes him an undesirable candidate. (Bruening 2018, 6)  
b. He [I-don't-care]-d his way out of the room.  
(Carnie 2000, 91)

This seems to be a clear example of syntax being 'visible' to morphology, since phrasal material can be used as input to a morphological process (suffixation).

Another apparent counter-example is the possibility of coordinating certain prefixes:

- (39) a. [pre- and even to some extent post]-war (economics)  
b. [pro- as opposed to anti]-war  
c. [hypo- but not hyper]-glycaemic (Spencer 2005b, 82)
- (40) a. [mono- and tri]-syllabic  
b. [pro- and en]-clitics  
c. [socio- and politico]-economic  
(Siegel 1974, 147, cited in Strauss 1982, 43)

In German, this also extends to verbal prefixes, leading to gapping constructions similar to (35):<sup>16</sup>

- (41) Peter be- und Maria ent-lud den LKW.  
*Peter BE- and Maria un-loaded the truck*  
'Peter loaded and Maria unloaded the truck.'

In all these cases, it seems that morphology is visible to syntax, since coordination is an operation in the phrasal syntax but here it is being applied to parts smaller than words.

Some have seen evidence such as this as damning. Marantz (1997, 207), for example, declares that "[l]exicalism is dead, deceased, demised, no more, passed on ...". All the same, more than

---

<sup>16</sup>My thanks to an anonymous reviewer for this example.

20 years later, Bruening has to again declare the lexicalist hypothesis “both wrong and superfluous” (the subtitle of Bruening 2018); apparently, earlier reports of lexicalism’s death were greatly exaggerated (to – further – misquote Mark Twain). In fact, the empirical situation is fraught, and none of the data presented in this section are unproblematic. For instance, Bruening (2018, 23–29) purports to explain the sub-lexical gapping and modification data in (35) and (36) in syntactic terms which make no reference to the notion of word, thus rendering LI superfluous. At the same time, the phenomenon of sub-lexical coordination illustrated in (39)–(40) is not at all as thoroughgoing as we might expect were morphology and syntax truly underpinned by exactly the same combinatory system. For although some English prefixes can be coordinated, others emphatically cannot:<sup>17</sup>

- (42) a. \*[un- or re]-tie  
b. \*[i{n|m}- or ex]-port (Spencer 2005b, 82)  
c. \*[ex- and se]-cretions (Siegel 1974, 147)

And it does not seem to be possible at all with suffixes:

- (43) a. \*fear-[some and -less]  
b. \*thought-[ful and -less]  
c. \*interest-[ed and -ing] (Strauss 1982, 43)

Lieber and Scalise (2007, 3) therefore express a sort of compromise position, admitting that LI cannot be valid in a strict sense, but viewing it as a kind of default or strong tendency: “we know that

---

<sup>17</sup> The distinction appears to be between what Siegel (1974) calls Class I and Class II prefixes. Class I prefixes are both morphologically and functionally ‘closer’ to the stem: they always appear nearer to the stem than Class II prefixes, for example, and unlike their Class II counterparts they can affect lexical stress assignment. It is perhaps unsurprising then that the prefixes in (42) that resist coordination are Class I – their closeness to the stem is reflected in their inability to be separated from it by a syntactic process like coordination. Note that the same class of prefixes can behave differently in different languages, however: the German *Im- und Export*, for example, is apparently impeccable. (One can also find hits online for *im- and export* in English, but many of these seem to be in forum posts written by German speakers: see, for example, <https://adobe.ly/3PoQKpo> or <https://bit.ly/42U4o75> [accessed June 22nd 2023].)

morphology and syntax interact, and that this interaction is not a one way affair: morphology sees syntax and syntax sees morphology. Nevertheless this two way interaction is highly constrained”. This is echoed more recently by Cappelle (2022, 204), who points out that “[a]ny randomly selected stretch of discourse is likely to prove that morphologically complex words stay together as undivided units and that they tend not to include any above-word-level components”.

Ultimately, a large part of the problem is this: deciding whether one or another piece of linguistic data is a LI violation depends hugely on one’s other theoretical assumptions (Desjardins 2023, 19–20), and so the enterprise of proving or disproving LI by looking for supportive or problematic constructions in the world’s languages is a largely hopeless one. There are, however, systematic differences between morphology and syntax at a higher level of abstraction that plead for a principled separation between the two.

Firstly, morphology applies strict ordering constraints on morphemes, even in languages where the syntax imposes no ordering constraints. For example, case markers and verbal inflection in Latin always follow the stem, even though any of the six permutations of the three *words* in (44) is grammatical:

- (44) a. mil-es        coqu-um laud-at.  
              *soldier-NOM cook-ACC praise-3SG.PRES.INDIC*  
              ‘The soldier praises the cook.’
- b. \*es-mil coqu-um laud-at
- c. \*mil-es um-coqu laud-at.
- d. \*mil-es coqu-um at-laud.

And, of course, morphemes from different words cannot be interleaved, even though discontinuous *constituents* are permitted (Snijders 2012, 2015, 211–212). That is, so-called ‘free word order’ languages are *not* ‘free morpheme order’ languages.

Secondly, and perhaps more foundationally, there are important differences in the computational complexities of morphology and syntax (Asudeh *et al.* 2013, 4–5). Morphology falls almost entirely within the class of finite-state languages (Roark and Sproat 2007, ch. 2),

with the sole exception being unbounded reduplication (Culy 1985).<sup>18</sup> Syntax, on the other hand, falls almost entirely within the class of context-free languages,<sup>19</sup> with the sole exception being cross-serial dependencies (Shieber 1985).<sup>20</sup> Assuming there is no formal difference between morphology and syntax, as CONSTRUCTIONS-ALL-THE-WAY-DOWN would have it, then this contrast is puzzling. As Asudeh *et al.* (2013, 5) put it:

[i]f morphology has the full power of syntax, why are there no clear morphological equivalents of unbounded or nested dependencies? [...] Similarly, why do we fail to find reduplication in the syntax, if there is no important formal distinction between morphology and syntax?

Note that the claim here is not merely that morphology is less powerful than syntax, but that the two systems are in fact *disjoint*: there are syntactic phenomena, like unbounded or nested dependencies, which we do not observe in morphology, and there are morphological phenomena, like reduplication or root-and-pattern, which we do not observe in syntax.<sup>21</sup>

I am not aware of any arguments in the literature which have addressed these concerns, and they seem to strongly suggest that we need to be able to distinguish between processes happening above the level of the word and processes below. Any framework that makes this

---

<sup>18</sup> See Wang and Hunter 2023 for a minimal extension to the class of regular languages designed to account for just this kind of pattern.

<sup>19</sup> See Partee *et al.* 1990, 480–482 for a proof that English is not a finite-state language, and see Gazdar *et al.* 1985 for a comprehensive syntactic theory which is nonetheless self-avowedly context free.

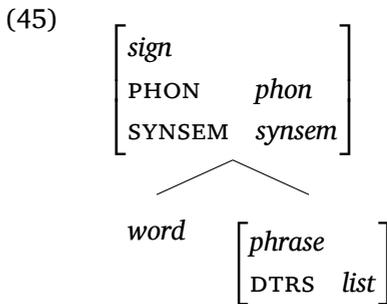
<sup>20</sup> Even though Shieber's findings show that the human language faculty must in general be capable of learning languages which are at least mildly context-sensitive in their strong generative capacity, further evidence of the necessity of greater-than-context-free power has not been abundantly forthcoming. It seems entirely plausible, as Pullum and Rawlins (2007, 285) opine, that languages simply vary in this respect. Perhaps non-context-freeness is a typological corner case, and designing our formalisms around it is merely generalising to the worst case.

<sup>21</sup> Contrastive Reduplication in English (*Do you LIKE HIM-like him?*) offers a potential counter-example to the idea that reduplication is not found in the syntax, since it has been claimed to obey (morpho)syntactic rather than purely prosodic constraints (Ghosheshi *et al.* 2004).

*impossible* ought to be treated with a degree of suspicion, therefore. So, is CxG such a framework?

In fact, the CxG position may have been overstated by its critics (or, depending on your point of view, the CxG position may be thought of as unclear/non-committal). On the same page that Goldberg (1995, 7) makes her oft-cited claim that “[i]n Construction Grammar, no strict division is assumed between the lexicon and syntax”, she goes on to clarify that “[i]t is not the case, however, that in rejecting a strict division, Construction Grammar denies the existence of any distinctly morphological or syntactic constraints (or constructions)”. It would seem our choice is not, therefore, between two extreme positions – on the one hand, a strict version of LI where syntax and morphology are computationally distinct processes, and, on the other, the obliteration of LI and total collapse of the syntax-morphology boundary. Rather, a third way is possible (and indeed espoused by Goldberg), where syntax and morphology operate under the same computational system, but where a formal distinction is drawn between morphemes, words, and phrases, meaning that linguistic processes can be sensitive to these contrasts (see Ackema and Neeleman 2004 for a similar approach outside of CxG).

This approach is readily embodied by any type-driven framework. For example, a standard HPSG type signature includes the sub-section shown in (45) (Przepiórkowski and Kupść 2006, §3.3):



Here we see that words and phrases are both sub-types of *sign*, and that what defines a sign is the pairing of PHONOlogical form with syntactic and semantic information (SYNSEM). That is, “both lexical and syntactic constructions are essentially the same type of declaratively represented data structure: both pair form with meaning” (Goldberg

1995, 7). But since *word* and *phrase* are still distinct types, it remains possible for certain constructions to be more restrictive: for example, ‘morphological’ constructions can be defined as those which require their mother to be specifically of type *word* (while ‘syntactic’ constructions require that their daughters (DTRS) merely be of type *sign*). I have illustrated this point with HPSG since it makes the cut so clearly and succinctly, but the same point could be made with HPSG’s explicitly constructionist cousin, Sign-Based Construction Grammar (SBCG: Boas and Sag 2012; Michaelis 2015),<sup>22</sup> which likewise recognises an early cleavage between lexical constructs and phrasal constructs (cf. Sag 2010, 499). In other words, even existing implementations of CxG do not take rejection of LI as a *sine qua non*.

CxG can therefore be made to fit with Lieber and Scalise’s (2007, 18) conclusion “that the interaction between word formation and syntax goes both ways, but that nevertheless it is quite restricted”: there are formalisms for CxG which do not in themselves preclude syntax-morphology interactions, but do give a means of restricting it and/or only permitting it on a construction-by-construction basis. What of LFG, then? LI must be weakened, it seems; but how easy is this to do?

As it happens, LFG already sanctions a weaker than strict interpretation of LI. In one common LFG formulation of LI, its scope is limited to c-structure:

(46) **Lexical Integrity:**

Morphologically complete words are leaves of the c-structure tree, and each leaf corresponds to one and only one c-structure node.

(Bresnan *et al.* 2016, 92)

That is, words are syntactic atoms when it comes to phrasal constituency, but when it comes to functional information, the internal morphological features of a word may very well be visible to syntax.

This view is well motivated, since there are numerous instances where what is expressed analytically in one language is expressed syn-

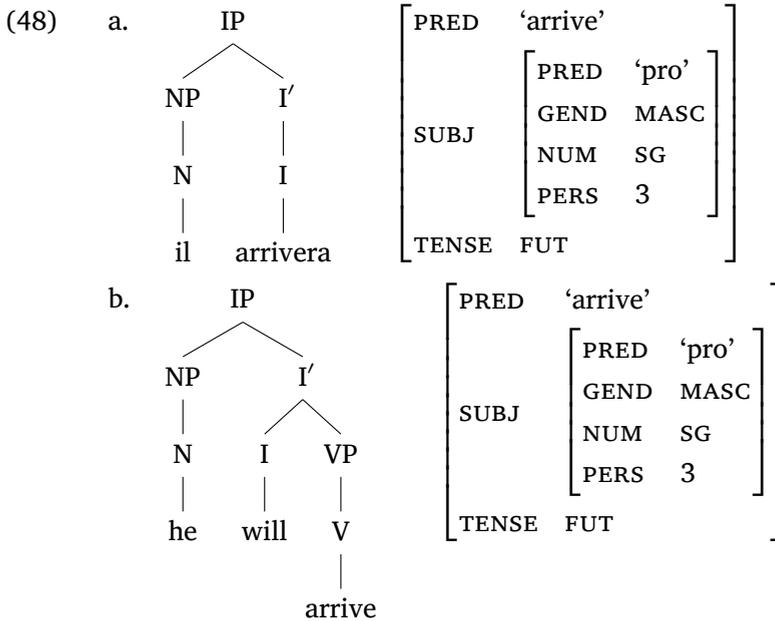
---

<sup>22</sup>Sag (2007, 403, 2010, 486) is explicit about SBCG being simply a variant of HPSG. In fact, HPSG is itself fundamentally constructionist, even though it does not bear the ‘construction grammar’ name (Sag 1997; Müller 2021).

thetically in another (Asudeh *et al.* 2013, 7–9). For instance, future tense in Romance languages can be expressed via verbal inflection, whereas English uses an auxiliary (Asudeh *et al.* 2013, 7):

- (47) a. Il arrivera. (French)  
*He arrive-3SG.FUT*  
 ‘He will arrive’  
 b. He will arrive.

While the c-structures of these sentences will differ, since the French tree has two terminal nodes and the English three, their f-structures will be identical:



In English, the attribute-value pair  $\langle \text{TENSE}, \text{FUT} \rangle$  is contributed by syntax, whereas in French, it is contributed by morphology.<sup>23</sup> In the latter case, the syntax, in the form of f-structure, can clearly ‘see’ the morphological features of words, even though c-structure is blind to morphological structure. That is, the syntax sees that *arrivera* contributes a

<sup>23</sup>An instance of the cross-linguistic phenomenon whereby, to use the LFG slogan, “morphology competes with syntax” (Bresnan 1998).

future tense feature, but it does not see that it is specifically the suffix *-a* which does so.

Thus, it is not true that morphology is wholly isolated from syntax in LFG, but it remains true that morphological *structure* is. In fact, this view is quite in keeping with one formulation of LI, that of Di Sciullo and Williams (1987, 49), whereby “words have ‘features’, or properties, but these features have no structure, and the relation of these features to the internal composition of the word cannot be relevant in syntax – this is the thesis of the atomicity of words, or the lexical integrity hypothesis, or the strong lexicalist hypothesis”. Falk’s (2001, 26) suggestion that “[l]exical integrity as understood by LFG [...] is a limited sort of lexical integrity” is thus perhaps a little premature, but does highlight that LI in LFG is limited to c-structure; by contrast, the contribution of a single word can be spread throughout f-structure, giving the impression of undermining said word’s atomicity. The orthodox view in LFG is therefore perhaps more in line with Cappelle’s (2022, 196) conclusion that LI should be reformulated “as a principle forbidding the manipulation of words, rather than access to word-internal structure” – ‘manipulation’ of words would correspond to splitting them up at c-structure.

As such, however, even this weakened version of LI would disallow the coordination of affixes seen in (39)–(40), and certainly the phrases hosting affixes seen in (38). So it may well be that LFG has to accept even greater concessions. There is some lurking awareness of this in the LFG literature. Analyses occasionally make use of ‘sub-lexical’ entries; that is, lexical entries for morphemes, written as if they were leaves in the c-structure tree, in clear violation of LI (e.g. King 1995; Nordlinger 1998; Marcotte 2009; Bresnan *et al.* 2016). Usually, it is implied (though often not stated explicitly) that these have no formal status, and should instead be viewed as descriptions of generalisations over lexical entries (of the sort that would nowadays be captured by templates), but sometimes suggestions are made to incorporate actual sub-lexical phrasal syntax (Marcotte and Kent 2010). Recent work in the LFG variant Lexical-Realizational Functional Grammar (L<sub>R</sub>FG: Melchin *et al.* 2020; Asudeh *et al.* 2021; Asudeh and Sidiqi 2022, to appear) takes this as its starting point, and assumes a rich sub-lexical syntactic structure, inspired by Distributional Morphology (Halle and Marantz 1993), thus rejecting LI wholesale. As

argued above, however, this may be throwing the baby out with the bathwater. Mainstream LFG has tended to adopt a halfway-house solution, using non-projecting categories (Toivonen 2003), which can be adjoined at the  $X^0$  level, to represent the kinds of elements that exist somewhere between word and morpheme. This treats LI as the default position, but allows a controlled relaxation of it in certain circumstances – such an approach has been used in the analysis of case (Spencer 2005a) and compounding (Lowe 2015), for example.

To sum up: the abolition of the distinction between word and phrase or morphology and syntax implied by a strict reading of CONSTRUCTIONS-ALL-THE-WAY-DOWN is not a necessary (or indeed empirically justified) tenet of CxG; at the same time, the *absolute* separation of the two implied by a strict reading of LI is not a necessary (or indeed empirically justified) tenet of LFG either. There is therefore ample room for common ground between the approaches, and we need not see the conflict between CONSTRUCTIONS-ALL-THE-WAY-DOWN and LI as a reason to dismiss LFG as a formalisation of CxG. But we are not home and dry yet! The proof of the pudding is in the eating, and the best way to validate a theoretical claim is to see it implemented. In the following section, therefore, I will demonstrate how LFG handles constructions. We will see that many formal idioms, including argument structure constructions, can be handled comfortably, and that the formalism actually accommodates divergent theoretical perspectives. However, when it comes to substantive idioms, we run into problems, and a change to the framework is needed.

#### 4

#### CONSTRUCTIONS IN LFG

So far, although I have argued that LFG has potential as a formal framework for CxG, we have not seen any examples of LFG analyses of constructional phenomena. This section will provide just that. We begin in Section 4.1 with Goldberg-style argument structure constructions, and show that LFG is compatible with either a lexical or constructional view of argument structure. Section 4.2 then shows how LFG can handle (some kinds of) formal idiom, but concludes that the presence of arbitrary phonological material (i.e. words) that

does not (appear to) contribute compositionally to the meaning of the construction causes problems: the more lexically filled a construction is, the more difficult it is for LFG to accommodate it. This is clearly most problematic for substantive idioms, which are the subject of Section 4.3.

*Argument structure constructions*

4.1

Recent work in LFG+Glue has treated argument structure not as a separate level of the projection architecture (as in e.g. Butt *et al.* 1997; Kibort 2007) but as a phenomenon at the syntax-semantics interface (Asudeh and Giorgolo 2012; Asudeh *et al.* 2014; Findlay 2016, 2020; Przepiórkowski 2017; *i.a.*). This research adopts a neo-Davidsonian event semantics (Parsons 1990), whereby the meaning contribution of a verb is not a predicate of or relation between individuals, but rather a predicate of eventualities (events or states) conjoined with a number of semantic role predicates that relate participants to that eventuality. For example, rather than the traditional meaning in (49), a verb like *sneezes* would have the meaning in (50):

(49)  $\lambda x.\text{sneeze}(x)$

(50)  $\lambda x.\lambda e.\text{sneeze}(e) \wedge \text{agent}(e, x)$

This opens up the possibility of splitting the valency information apart from the lexically-specified eventuality predicate, as in (51):

(51) a.  $\lambda e.\text{sneeze}(e)$

b.  $\lambda P.\lambda x.\lambda e.P(e) \wedge \text{agent}(e, x)$

The result of applying (51b) to (51a) is (50), but by factoring out these two components of meaning we have separated out the core lexical meaning from what would be seen in CxG as the constructionally-provided argument structure meaning (Goldberg 1995). This means that the same core lexical meaning can be used across diathesis alternations (Asudeh and Giorgolo 2012) or other argument structure frames (Asudeh *et al.* 2014).

Of course, in Glue Semantics these meaning terms are paired with a linear logic type which anchors them in the syntax:

- (52) a.  $\lambda e.\mathbf{sneeze}(e) : v_{\uparrow} \multimap t_{\uparrow}$   
 b.  $\lambda P.\lambda x.\lambda e.P(e) \wedge \mathbf{agent}(e, x) :$   
 $(v_{\uparrow} \multimap t_{\uparrow}) \multimap e_{(\uparrow\text{SUBJ})} \multimap v_{\uparrow} \multimap t_{\uparrow}$

Using  $v$  as the type of events, we can see that the meaning constructor in (52b) consumes the meaning constructor in (52a) to produce a dependency on the verb's subject.

We can combine the core meaning with other valency templates to produce other constructional meanings. For instance, we can represent the English caused-motion construction (Goldberg 1995, ch. 7), exemplified in (53), with the meaning constructor in (54):<sup>24</sup>

(53) Frank sneezed the tissue off the table.

- (54)  $\lambda P.\lambda x.\lambda y.\lambda Q.\lambda e.P(e) \wedge \mathbf{agent}(e, x) \wedge \mathbf{theme}(e, y) \wedge \mathbf{goal}(e, Q) :$   
 $(v_{\uparrow} \multimap t_{\uparrow}) \multimap$   
 $e_{(\uparrow\text{SUBJ})} \multimap e_{(\uparrow\text{OBJ})} \multimap (e_{(\uparrow\text{OBL})} \multimap t_{(\uparrow\text{OBL})}) \multimap v_{\uparrow} \multimap t_{\uparrow}$

This will require that the verb be accompanied by an OBJECT and an OBLIQUE in the syntax. If these dependents are not present, this meaning constructor will be unusable, since there will be no meaning constructors which match the types required. (We return to this point momentarily.)

For the sake of brevity, let us name our two argument structure frames using templates:

- (55) AGENT-FRAME :=  
 $\lambda P.\lambda x.\lambda e.P(e) \wedge \mathbf{agent}(e, x) :$   
 $(v_{\uparrow} \multimap t_{\uparrow}) \multimap e_{(\uparrow\text{SUBJ})} \multimap v_{\uparrow} \multimap t_{\uparrow}$
- (56) CAUSED-MOTION-FRAME :=  
 $\lambda P.\lambda x.\lambda y.\lambda Q.\lambda e.P(e) \wedge \mathbf{agent}(e, x) \wedge \mathbf{theme}(e, y) \wedge \mathbf{goal}(e, Q) :$   
 $(v_{\uparrow} \multimap t_{\uparrow}) \multimap$   
 $e_{(\uparrow\text{SUBJ})} \multimap e_{(\uparrow\text{OBJ})} \multimap (e_{(\uparrow\text{OBL})} \multimap t_{(\uparrow\text{OBL})}) \multimap v_{\uparrow} \multimap t_{\uparrow}$

One possibility is to associate these with the verb *sneezes* in the lexicon, as shown in (57). This represents what Müller and Wechsler

<sup>24</sup>I assume the second argument of the *goal* predicate is a relation expressing a location, e.g.  $\lambda x.\mathbf{off}(x, \iota y[\mathbf{table}(y)])$  in this case, and therefore has a functional type in the linear logic.

(2014) call a lexical approach to argument structure, as opposed to the phrasal, or constructional, approach of e.g. Goldberg (1995), and which we discuss below.

- (57) sneezes V (↑ PRED) = ‘sneeze’  
 $\lambda e.\text{sneeze}(e) : v_{\uparrow} \text{---} t_{\uparrow}$   
 { @AGENT-FRAME |  
 @CAUSED-MOTION-FRAME | ... }

The third clause in (57) expresses a disjunction, and is intended to include all the other potential argument structure constructions that a verb like *sneezes* might enter into. Such disjunctions can themselves be encoded in templates which name different types of verb, for example, thus allowing generalisations to be captured (of the type that would be captured by lexical rules in other approaches, or indeed in earlier LFG analyses).

Recall that if we choose the CAUSED-MOTION-FRAME, the verb must be accompanied by an object and an oblique argument. If these dependents are not present in the syntax, we will have a situation of RESOURCE DEFICIT in the semantics (Asudeh 2012) – there will be too few meaning constructors for the valency frame constructor to consume, and so no valid proof for the sentence. This results in ungrammaticality. The effect of this is that the various argument structure constructions are only licensed when the verb is in the correct syntactic environment, but this is achieved without actually placing any constraints on the syntax: the constraints are instead on the syntax-semantics interface.

In addition to the lexical approach, LFG is also compatible with the alternative, constructional view, whereby the argument structure frames are associated not with the lexical entries of verbs, but with particular phrasal configurations, as illustrated in (58) and (59). This is because LFG annotations can be added to phrase-structure rules just as well as to lexical entries (since formally lexical entries just *are* phrase-structure rules).

- (58) IP → NP I'  
 (↑ SUBJ) = ↓ ↑ = ↓  
 { @AGENT-FRAME | ... }

$$(59) \quad \text{VP} \rightarrow \begin{array}{c} \text{V}' \\ \uparrow = \downarrow \\ \{\text{@CAUSED-MOTION-FRAME} \mid \dots\} \end{array}$$

Notice that we are still underspecifying the phrase structure associated with these templates; for example, CAUSED-MOTION-FRAME will still be present on a V' node even when it does not have the correct number of daughters, viz. an object and oblique alongside the verb. This is possible because of the disjunctive approach, which relies on resource sensitivity to select only the appropriate meaning constructor(s). But of course nothing stops us associating the constructional meaning with more specific phrase-structure rules either, if this is preferred for theoretical reasons:

$$(60) \quad \text{V}' \rightarrow \begin{array}{ccc} \text{V} & \text{NP} & \text{PP} \\ \uparrow = \downarrow & (\uparrow \text{OBJ}) = \downarrow & (\uparrow \text{OBL}) = \downarrow \\ \{\text{@CAUSED-MOTION-FRAME} \mid \dots\} & & \end{array}$$

Much ink has been spilled on the question of whether argument structure is best analysed as a lexical or constructional/phrasal phenomenon (see, among others, Goldberg 1995, 2006, 2013; Müller 2002, 2006, 2018; Tomasello 2003; Goldberg and Jackendoff 2004; Müller and Wechsler 2014). In particular, Müller (2018) provides a detailed critique of exactly the phrasal LFG approach sketched above, highlighting numerous empirical problems. It may well be the case that argument structure should be handled lexically, therefore; but it is also true that at present most CxG approaches do not do this, and instead take the phrasal view, following Goldberg (1995). LFG as a formalism is thus capable of expressing the canonical CxG view, even though we might ultimately reject such a view on empirical grounds. At the same time, though, LFG also provides for the possibility of a lexical analysis (or, in fact, what Goldberg 2013, 447–448 calls a “derivational verb template” analysis). The main point of this section is therefore that the *formalism* of LFG offers the analytical flexibility to make the choice about argument structure either way, depending on which *theoretical* stance one takes. Indeed, and in keeping with the CxG focus on diversity and variety in linguistic phenomena, the LFG formalism in fact allows us to allocate argument structure meanings lexically or constructionally *on a case by case basis*, thus offering a more

empirically responsive, and perhaps less ideologically driven, kind of theorising.

*Formal idioms*

4.2

These same techniques can be applied quite liberally to all manner of constructional meanings. For example, Asudeh *et al.* (2013) give a very detailed analysis of related ‘traversal’ constructions in Swedish, English, and Dutch, illustrated in (61) by the English *way*-construction:

- (61) a. Sarah elbowed her way through the crowd.  
       b. Sarah whistled her way across the room.

(Asudeh *et al.* 2013, 12)

This has the special meaning that Sarah traversed the crowd/room, and that either the means (as in (61a)) or the manner (as in (61b)) of this traversal was the activity described by the main verb. None of the words in (61) normally conveys this meaning alone, so it seems to emerge from the construction itself.

However, Asudeh *et al.* (2013) argue that the constructional meaning need not be hosted by a phrasal configuration, since in English there is nothing special about the syntax of the *way*-construction. As in (61), it employs a standard [V NP PP] configuration, which is witnessed by many other constructions. Rather, what is special about the *way*-construction is the obligatory presence of the word *way* – Asudeh *et al.* (2013, 30) therefore choose the lexical entry for this word as the host of the constructional meaning (highlighted here with a box):<sup>25</sup>

- (62) way        N    (↑ PRED) = ‘way’  
                                    $\lambda x.\mathbf{way}(x) : e_{\uparrow} \multimap t_{\uparrow}$   
                                   (**@ENGLISH-WAY**)

<sup>25</sup>When reproducing formal analyses from Asudeh *et al.* 2013, I omit some detail to avoid unnecessary exposition, and modify some expressions to bring them into conformity with the choices made in this paper. This does not affect any of the arguments made here, but the reader should be aware of the discrepancies, and should consult Asudeh *et al.* 2013 for the formal details.

This is the normal lexical entry for run-of-the-mill *way*, with the addition of an optional constructional meaning constructor (optionality is indicated by surrounding a piece of description in parentheses), abbreviated by the template ENGLISH-WAY. I will not unpack this meaning here, since the higher-level principle can be grasped without going into the details of the analysis, but it adds the additional traversal meaning to the clause, and makes the verb of which *way* is the object either the means or the manner of this traversal event. Note that the normal meaning of *way* is not optional in (62), and therefore survives in the constructional use too; in fact, it is equated with the **path** through which the traversal event proceeds (Asudeh *et al.* 2013, 30–31), enabling a straightforward analysis of instances where *way* is modified or possessed by something other than the subject (Asudeh *et al.* 2013, 13):

- (63) a. As ambassador, Chesterfield negotiated [Britain’s way] into the Treaty of Vienna in 1731.  
b. In these last twenty years Richard Strauss has flamed [his meteoric way] into our ken – and out of it.

The Swedish traversal construction analysed by Asudeh *et al.* (2013), called the directed motion construction (DMC) by Toivonen (2002), is illustrated in (64):

- (64) a. Sarah armbågade sig genom mängden.  
*Sarah elbowed SELF through crowd.DEF*  
≈‘Sarah elbowed her way through the crowd.’  
(Asudeh *et al.* 2013, 13)
- b. Han ljög sig ut ur armén.  
*he lied SELF out of army.DEF*  
≈‘He lied his way out of the army.’ (Toivonen 2002, 315)

This differs from the English construction in that it only has a means (not a manner) interpretation, and that there is no equivalent of *way*, i.e. a fixed word which is always present. Although the presence of the (simplex) reflexive, here illustrated by *sig*, is obligatory, its form will change depending on the person and number of the subject, with which it agrees (e.g. it will be *mig* for a 1SG subject, *dig* for 2SG, etc. – see Toivonen 2002, 322). Asudeh *et al.* (2013) therefore suggest that

the constructional meaning here *should* be associated with a particular phrasal schema (Asudeh *et al.* 2013, 22):

$$\begin{array}{rcccl}
 (65) & V' \rightarrow & V & & NP & & PP \\
 & & \uparrow = \downarrow & & (\uparrow \text{OBJ}) = \downarrow & & (\uparrow \text{OBL}) = \downarrow \\
 & & & & (\downarrow \text{PRONTYPE}) = \text{SIMPLEX-REFL} & & \\
 & & & & \boxed{\text{@SWEDISH-DMC}} & & 
 \end{array}$$

Once again, note that all of the normal meanings for the words involved in the DMC persist in the constructional meaning. The construction introduces additional meaning, in the form of a new traversal event etc., but does not replace any existing meanings.

Just as we saw above with argument structure constructions, the approach of Asudeh *et al.* (2013) illustrates the analytical flexibility that LFG affords researchers: constructions, in the theory-neutral sense, can be given either a lexical or phrasal analysis, depending on (i) the details of the construction itself and/or (ii) broader theoretical concerns (or preferences). For instance, it would be wholly possible to associate the ENGLISH-WAY template, and its constructional meaning, with a special phrase-structure rule just like in Swedish, rather than hosting it in the lexical entry for *way*. The only substantial change would be the addition of a requirement that the NP the template annotates have the PRED value ‘way’, since, unlike Swedish, there is a specific lexical element which is obligatory in the English construction. This would arguably be a less direct way of encoding such a requirement, but the important point is that the formalism leaves one free to make such decisions on theoretical grounds alone – no choice is imposed by the framework.

So far so good, then! We have seen that LFG has tools at its disposal which enable it to handle constructional phenomena. However, what these constructions all have in common is that they involve *extra* meaning being added on top of the standard, literal meanings of their parts. Sometimes meanings are also ‘realigned’, e.g. the main verb of the *way*-construction is relegated from expressing the main predicate of the clause to merely expressing the means or manner of the traversal event, but none are discarded. Indeed, the resource sensitivity of Glue Semantics makes this quite difficult to do. But plenty of constructions have meanings that do not merely make unconven-

tional use of the meanings of their parts, but actually override or ignore them.

For example, in the *WXDY* construction, illustrated in (66), both *what* and *doing* do not contribute their usual semantic content: the construction is not asking for the identity of an activity being undertaken – indeed, there need not be any ‘doing’ happening at all (this is especially clear when the subject is inanimate, as in (66b) and (66c)).

- (66) a. What are your children doing playing in my garden?  
b. What do you think your name is doing in my book?  
c. I wonder what the salesman will say this house is doing without a kitchen.  
d. What’s a nice girl like you doing in a place like this?

(Kay and Fillmore 1999, 3, 5)

It seems the only way to give a satisfactory LFG analysis of this construction would be to have special versions of *what* and *doing* which either contribute no meaning or contribute some part of the overall constructional meaning instead of their usual semantic content.<sup>26</sup> Such a move may be empirically adequate, but it rather flies in the face of CxG assumptions, since now the construction is distributed through the lexicon and grammar rather than being represented in one place: even if the whole constructional meaning can be encoded in a single template on a special phrase-structure rule, we still need to have two new lexical entries for the special versions of *what* and *doing*.

---

<sup>26</sup> Equivalently, one could stick to a single lexical entry for each word, but give a disjunctive specification of its meaning, with the ‘empty’ or construction-specific meaning as one of the disjuncts (see also footnote 30 below). Alternatively, one might imagine keeping the regular lexical entries but using specially-designed, construction-specific meaning constructors to ‘throw away’ the literal meanings (cf. the *MANAGER RESOURCES* of Asudeh 2012, 128–134) – this is the approach proposed by Arnold (2015) for idioms like *kick the bucket*. However, such a strategy is ultimately untenable, since it makes radically incorrect predictions about modification (specifically, that it should be possible to vacuously modify words whose meanings are removed in this way – e.g. *What’s a nice girl like you doing carefully in a place like this?* ought to have the same meaning as (66d)); see Findlay 2017, 228–229 for more details.

And this will only multiply as more constructions are considered. (See Section 4.3.2 for further discussion.)

So, we can conclude that LFG is well suited to handle highly schematic idioms, since these involve overlaying additional meaning on existing lexical resources, but that as constructions become less schematic and more substantive, problems begin to arise. In the next section, I examine some attempts to confront these challenges, and suggest a different solution.

### Substantive idioms

4.3

Alongside intermediate constructions like *WXDY*, there are fully substantive idioms like the following:

- (67)
- a. Don't worry; we still have our **ace in the hole**.  
(*ace in the hole*  $\approx$  '(hidden) resource or advantage')
  - b. Chrisjen likes to be kept **in the loop**.  
(*in the loop*  $\approx$  'informed (about a particular matter)')
  - c. **Pull yourself together**, man! We're not giving up that easily!  
(*pull oneself together*  $\approx$  'calm down/compose oneself')
  - d. These new import regulations really **take the biscuit**.  
(*take the biscuit*  $\approx$  'be especially egregious/shocking/annoying')
  - e. If you **let the cat out of the bag** too early there will be trouble.  
(*let the cat out of the bag*  $\approx$  'reveal the secret')

In these kinds of idioms, the literal meanings of the words involved simply do not appear – they are replaced wholesale by different, idiomatic meanings. This is a major problem for the approach of Asudeh *et al.* (2013). *Taking the biscuit*, for instance, involves no taking event and no contextually salient biscuit, but unless we do something to prevent it, precisely these meanings will be introduced by the standard lexical entries for *take*, *the* and *biscuit*. So even if we also introduce a meaning for 'be especially egregious' via some constructional template (associated with a phrase-structure rule or with one or more of the words themselves), we still need to do something with the 'left

over' literal meanings – that is, we find ourselves in a state of RESOURCE SURPLUS: there will be no way to successfully incorporate these meanings into the linear logic proof, and so the sentence will be ruled out by the grammar.

There is also the secondary challenge of ensuring the idiomatic meaning only arises when all of the required words appear in the correct configuration. This is precisely what makes these idioms substantive: their parts cannot be switched out, even for semantically very similar constituents. E.g. *in the ring* does not have the idiomatic meaning of *in the loop*, nor *yank oneself together* the idiomatic meaning of *pull oneself together*.<sup>27</sup>

The fundamental problem is that the only level at which phonological form is paired with meaning in LFG is in the lexicon, and the lexicon contains only words.<sup>28</sup> As we saw in the previous section, abstract phrasal configurations can also be paired with meanings, and words can be associated with complex meanings that reference structurally distant parts of the phrase, both of which give LFG the appearance of licensing phrasal constructions more generally. But there is no way of assigning specific meanings to structurally complex *multiword* units, and this is why substantive idioms are challenging. In this section, I will present two potential solutions to this problem, before advocating for a third way, more in keeping with the intuitions of CxG.

---

<sup>27</sup> There are, however, cases where substantive idioms are distorted for communicative effect, e.g. using *shatter the ice* as an intensified version of *break the ice* 'remove or lessen the tension at a first meeting' (McGlone *et al.* 1994). See Findlay 2019, 43–47, 84–87, 92–96, 321–335 and references therein on lexical flexibility in substantive idioms more generally and on metaphorical extensions to idioms.

<sup>28</sup> This is not exactly true of the LFG formalism *stricto sensu*, since there is no obstacle to writing phrase-structure rules whose right-hand sides contain a mixture of terminals and non-terminals. In practice, however, this does not happen, presumably because it runs contrary to the lexicalist style of theorising. It is important to note, though, that such mixed rules still only permit description of a single generation in the tree, i.e. the daughters of a single mother node, so it does not make it possible to associate phonological form and semantic content across truly unbounded spans of c-structure – on which see Section 4.3.3.

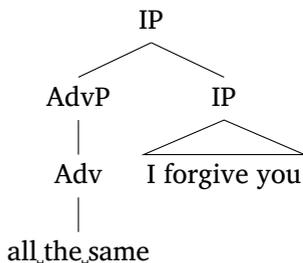
Since the only locus of phonological form-meaning pairing in LFG is the lexicon, one very simple solution to the problem of substantive idioms is to deny their multiword status and instead treat them as “words with spaces” (Sag *et al.* 2002), so that they can be given lexical entries. This is probably the correct analysis for lots of the more morphosyntactically rigid idioms, i.e. what Sag *et al.* (2002) call FIXED EXPRESSIONS – those whose parts do not inflect and cannot be manipulated by syntactic processes – such as the examples in (68):

- (68) a. all the same  
 b. by the by  
 c. in short  
 d. no can do

These can be represented in the grammar as if they were single words (Dyvik *et al.* 2019), i.e. single c-structure terminals, since they cannot be interrupted or split up (i.e. they obey the strictures of LI):

- (69) all\_the\_same Adv (↑ PRED) = ‘all-the-same’  
*etc.*

- (70) All the same, I forgive you.



Indeed, there exist several expressions which are superficially very similar to these but which are written without spaces, and therefore already treated as single words:

- (71) a. although (cf. all though)  
 b. nonetheless/nevertheless (cf. none/never the less)  
 c. notwithstanding (cf. not withstanding)

If the difference here is purely an accident of orthography, then we are right to treat the expressions in (68) analogously, as single words.

Of course, many substantive idioms are not so fixed as to be amenable to an analysis as single words. This is clear from the fact that their sub-parts can be inflected, modified, and manipulated syntactically (which can lead to discontinuous linearisation):

- (72) INFLECTION:  
That takes/took/has taken/will take the biscuit.
- (73) MODIFICATION (Findlay 2017, 212):
- a. Musicians keep composing songs 'til they [kick the proverbial bucket].  
(*kick the bucket*  $\approx$  'die')
  - b. Britney Spears [...] [came apart at the mental seams].  
(*come apart at the seams*  $\approx$  'to be in a very bad state; to fall to pieces')
  - c. Maybe by writing this book I'll offend a few people or [touch a few nerves].  
(*touch a nerve*  $\approx$  'annoy/upset someone by referring to a sensitive topic')
  - d. Tom won't [pull family strings] to get himself out of debt.  
(*pull strings*  $\approx$  'exploit connections')
- (74) SYNTACTIC MANIPULATION:
- a. They finally [spilled the beans].  
(*spill the beans*  $\approx$  'reveal the secret(s)')
  - b. [The beans] were finally [spilled].  
(passivisation)
  - c. I really want to see [the beans] he [spills] under oath.  
(relativisation)
  - d. But [which particular beans] did he [spill]?  
(*wh*-fronting)

Each of these phenomena would be extreme violations of LI if the expressions in question were really single words.

It is this kind of data which motivates the CxG view that constructions (which can be of any size), not words, are the real building blocks of the grammar. Nonetheless, much recent theoretical work

on idioms has sought to avoid this conclusion and instead view idioms as licensed lexically.<sup>29</sup> In these theories, substantive idioms are treated as being made up of special versions of the words they contain, whose meanings combine to give an appropriate meaning for the whole construction. For example, there will be a special version, or a special sense, of *spill* that means ‘reveal’, and a special version or sense of *beans* that means ‘secrets’, so that the meaning of *spill the beans* can be ‘reveal the secrets’. We will call this the LEXICAL AMBIGUITY approach (hereafter ‘LA’; cf. Findlay 2017, 213), since it (i) treats idioms lexically, and (ii) does so by introducing ambiguities (e.g. *beans* is now ambiguous between literal ‘beans’ and idiomatic ‘secrets’).<sup>30</sup>

This kind of theory naturally explains the data in (72)–(74): it is no surprise that the parts of such expressions can be manipulated or modified, since they are just ordinary words, with their own meanings. It also explains why these expressions inhabit ordinary syntactic structures (like a verb+object VP).

However, this approach faces a number of issues (see Findlay 2019, 58–77 for a detailed critique). Firstly it must address the “collocational challenge” (Bargmann and Sailer 2018, 12): if *beans* can mean ‘secrets’ in *spill the beans*, how do we stop it from having this meaning elsewhere?

- (75) #Have you heard the beans?  
(≠ ‘Have you heard the secrets?’)

---

<sup>29</sup>This trend can be observed in many different frameworks, including, surprisingly, those which are, or have the potential to be, constructional: Sailer 2000 in HPSG, Kay *et al.* 2015 in SBCG, Lichte and Kallmeyer 2016 in LTAG, and Arnold 2015 in LFG. See also Bargmann and Sailer 2018. An early computational approach can be found in Fischer and Keil 1996.

<sup>30</sup>Lichte and Kallmeyer (2016) draw a distinction between LA approaches that invoke what they call a syntactic ambiguity, i.e. those which treat the different meanings of idiom words (like *spill* or *beans*) as belonging to different lexemes, and LA approaches that invoke what they call a semantic ambiguity, i.e. those which treat the idiom words as single, polysemous lexemes, rather than collections of homonymous ones. While most LA approaches take the former, syntactic, view, and so this is what I present in the text, Lichte and Kallmeyer (2016) advocate for the latter, semantic, view, and argue that it has formal, empirical, and psycholinguistic advantages. *Mutatis mutandis*, the challenges I note below still apply to this polysemy-based version of LA, however.

Although this may ultimately be surmountable through the use of (fairly extreme) selectional restrictions (see e.g. Sag 2007; Kay *et al.* 2015 for examples), it is not as straightforward as might be assumed, especially when parts of idioms can be separated across clausal boundaries and may not bear any direct syntactic relationship to each other, as is the case with the relativisation example in (74c) (Findlay 2017, 214–215). In this sentence, *spill* and *beans* bear no direct syntactic relationship to one another; instead, the relationship between them is mediated via a (in this case unpronounced) relative pronoun (see Dalrymple *et al.* 2019, 665–671 for the LFG analysis of relative clauses).<sup>31</sup>

Secondly, although LA makes sense for so-called decomposable idioms (what Nunberg *et al.* 1994 call **IDIOMATICALLY COMBINING EXPRESSIONS**), where the meaning of the idiom can be distributed among its parts, it is much less clear what the motivation might be for applying it to non-decomposable ones (what Nunberg *et al.* 1994 call **IDIOMATIC PHRASES**), where this is not the case. For example, we can well imagine a special meaning for *pull* such that it means ‘exploit’ and *strings* such that it means ‘connections’, which explains why (73d) has the meaning it does: *family* modifies the meaning ‘connections’ of *strings*, so that the sentence means that it is family connections which Tom refuses to exploit. But now consider idioms like *kick the bucket* (≈ ‘die’) or *shoot the breeze* (≈ ‘chat’). In neither case can we readily assign meanings to the parts individually; rather, the complex whole has a simplex meaning – expressible by a single word in English.<sup>32</sup> Only one word need host the meaning, therefore, and it is a wholly arbitrary decision which one we choose.<sup>33</sup> Perhaps we assign the head

---

<sup>31</sup> Falk (2010) advocates on the basis of such problematic data for eliminating this “mediated” analysis of relative clauses altogether, though this proposal has not been widely adopted, perhaps owing to the increased formal complexity it introduces.

<sup>32</sup> Of course, in keeping with the Principle of No Synonymy (Bolinger 1968; Goldberg 1995, 67), these paraphrases are necessarily inexact. *Kick the bucket* possesses entailments lacking in *die*, for instance, such as punctuality (*#she lay kicking the bucket for months* is decidedly infelicitous when compared with *she lay dying for months*); and *shooting the breeze* refers to a particular kind of aimless chit-chat, not just chatting in general.

<sup>33</sup> A reviewer points out that the “idiomatic mirroring” approach of Lichte and Kallmeyer (2016) alleviates this problem: since the meaning of a non-

*kick* the meaning ‘die’, and then have versions of *the* and *bucket* which make no semantic contribution at all (or at most a vacuous one). But we could equally well assign the ‘die’ meaning to *the* or *bucket* and then have the others words empty of meaning (see Findlay 2019, 67–74 for a discussion of these and various other possibilities explored in the LA literature). For non-decomposable idioms, LA is purely a technical solution, and no longer has any empirical motivation – unlike for decomposable idioms, where the parts were individually modifiable.

Thirdly, this strategy will lead to a massive proliferation of semantically empty lexical items – the lexicon will expand by as many entries as there are words in substantive idioms. This is because each semantically empty word must have its distribution constrained to a particular idiom so as to meet the collocational challenge. But this means the semantically empty *the* of *kick the bucket*, constrained to appear as the specifier of idiomatic *bucket*, cannot be the same as the semantically empty *the* in *shoot the breeze*, and so on.<sup>34</sup> Note that the situation is dif-

---

decomposable idiom is assigned to *all* of its parts, there is no decision to be made about where it should live. This is true, but does not mean their proposal escapes the charge of arbitrariness, for it shares with any LA approach the lack of independent motivation in the case of non-decomposable idioms. The only reason to assume that *bucket* means ‘die’ or that *breeze* means ‘chat’ is that they appear in larger phrases which have those meanings; there is no theory-external reason to believe they bear those meanings independently. Lichte and Kallmeyer (2016, 124–125) suggest that the existence of the expression *bucket list* shows that *bucket* in fact *does* have this meaning independently, but Findlay (2019, 70–71) gives multiple reasons to doubt this. A limited analogical creation such as this does not show that there is a word *bucket* meaning ‘die’ that has an independent existence outside the idiom; the parts of such expressions cannot freely be used productively, for instance: there is no *#bucket book* in which to write one’s bucket list, nor a *#breeze room* where one could shoot the breeze with a friend, etc. The idiomatic mirroring of Lichte and Kallmeyer (2016) is therefore just as guilty as any other LA approach of assigning meanings to words for no reason other than to serve the theory.

<sup>34</sup> Other approaches avoid having a proliferation of semantically empty words, but still face a blow-up in the lexicon due to the problem of properly constraining the distribution of idiom words. Bargmann and Sailer 2018 is one such example; see Findlay 2019, 71–73 for discussion. In the “idiomatic mirroring” approach of Lichte and Kallmeyer (2016), the lexicon is not expanded in quite the same way, since the difference between idiom words and their regular counterparts is treated as one of polysemy rather than homonymy, meaning there is just a single

ferent from that of e.g. expletive *it* or *there* in English (used in weather expressions like *it rains*, existential constructions like *there is/are*, and raising constructions like *it seems that* or *there seem(s) to be*), where we are happy to posit just a single lexical entry for each. This is because *it* and *there* occupy argument positions, and so their distribution will be constrained by standard well-formedness conditions, e.g. Completeness and Coherence in LFG (Dalrymple *et al.* 2019, 50–53).<sup>35</sup> By contrast, the grammar will freely permit NPs with and without determiners, which means that if we have an unconstrained semantically empty *the*, an NP like (76) will be ambiguous between a definite reading, where *the* has its usual semantic value, and a bare plural (generic) reading, equivalent to (77), where *the* is semantically inert:

(76) The students (are hard-working.)

(77) Students are hard-working.

Completeness and coherence cannot help us here, since we are dealing with the internal structure of an argument NP, rather than the presence or absence of an argument. Besides, we want the grammar to license both the *strings* in (76) and (77), it's just that the determiner in (76) must be the contentful one. This is why any hypothesised semantically empty *the* must have its distribution restricted to the idiom

---

lexical entry for each word. However, this approach is not thereby off the hook: while it may not explode the lexicon, it avoids this by pushing the complexity into the individual lexical entries. So while it's true that there need not be as many new lexical entries as there are words in substantive idioms, there will instead be as many new *senses* as there are words in substantive idioms. This means that common words like *the* will still be a problem, since, by idiomatic mirroring, they must possess a different meaning for each substantive idiom which they are part of, and so they will become massively ambiguous. That is, in addition to its literal meaning, the word *the* must also mean 'die' (*kick the bucket*), 'chat' (*shoot the breeze*), 'get angry' (*fly off the handle*), 'sleep' (*hit the hay*), etc., etc. It is certainly not apparent *a priori* that this situation is preferable to the constructional view which stores the idiom in a single place, and thus only expands the lexicon by as many entries as there are idioms.

<sup>35</sup>Constraining the distribution of expletives is one area where the resource sensitivity of Glue cannot straightforwardly replace syntactic constraints on valency like Completeness and Coherence. See Asudeh 2012, 113 for some discussion of this problem.

it is associated with, and this leads to the lexical explosion described above.

Findlay (2019, 74–76) discusses several more problems for LA, including difficulties with syntactically idiosyncratic expressions, and incompatibility with psycholinguistic evidence. But in fact what I consider the most damning objection is this: LA does not capture (in fact rejects) the most significant fact about substantive idioms – namely, that they have an ontological status as wholes. As Williams (2007) somewhat sardonically puts it, “[a] traditional view of idioms is that they are ‘things’, that is, linguistic units”. But LA bends over backwards to deny this: substantive idioms have no status as linguistic units; instead, they are conspiracies of single words. In a framework like LFG, where the only pairings of phonological form and semantic content allowed are words, this is the only strategy available. But such a strategy is flagrantly opposed to the CxG view of idioms, making this the real obstacle to considering LFG a suitable formalism for CxG.

#### Constructional LFG

#### 4.3.3

If substantive idioms are to be ‘things’ in the grammar, we need to have a way of listing them. At present, LFG cannot do this, since phrase-structure rules, the basic building blocks of the grammar (of which lexical entries are a subset), only describe c-structure relations between a mother and her daughters, nothing more remote. But of course, “constructions need not be limited to a mother and her daughters, but may span wider ranges of the sentential tree” (Fillmore *et al.* 1988, 501), so in order to adequately describe constructions, something needs to change.

There is another reason to reach the same conclusion, from an LFG-internal perspective. At present, LFG lexical entries include a functional description which gives details of all levels of structure *except* c-structure. This description can also include very long-distance relations within or between structures expressed by functional uncertainty paths. But descriptions of c-structure are limited to mother-daughter relations: the lexical entry identifies the category of the pre-terminal node which hosts it (i.e. its mother) and nothing else. Such a discrepancy is striking, and it is natural to want to remedy it.

This limitation comes from the decision to use a context-free grammar (CFG) for the description of c-structure. While this has

practical benefits in terms of parsing, it limits the expressive power of c-structure descriptions to this smaller, mother-daughter, domain of locality. Findlay (2019, ch. 5, to appear) therefore proposes to replace the CFG with something more expressive, namely a tree-adjoining grammar (TAG: Joshi *et al.* 1975; Joshi and Schabes 1997; Joshi 2005; Kallmeyer 2010, ch. 4). Although this increases the expressive power of c-structure descriptions, it does not alter the computational complexity of the LFG formalism as a whole: LFG already has more than context-free power (Berwick 1982; Nakanishi *et al.* 1992), and, even at its most constrained, is still slightly more powerful than a TAG (being equivalent to a LCFRS – see Wedekind and Kaplan 2020). By moving to a more expressive tree formalism, LFG can, however, more completely embrace the CxG assumption of EDL.

Findlay's (2019, ch. 5) proposal employs a description-based TAG (Vijay-Shanker 1992) and makes use of lexical entries which contain descriptions of the tree corresponding to the maximal functional projection of the lexical item, as is standard in LTAG (Schabes *et al.* 1988) – e.g. nouns are represented as NPs, but verbs are represented as clausal trees containing positions for their arguments. Parsing, in this version of LFG, consists of gathering up all the descriptions associated with the lexical items in a sentence, and then finding the minimal structures – *including* c-structure – which jointly satisfy them.

Under this view, lexical entries are descriptions, i.e. lists of constraints, which cover all levels of the projection architecture simultaneously. Of course, there is now no requirement that such lexical entries describe only a single word, or indeed that they describe *any* word – the objects we are talking about are simply descriptions of pieces of linguistic structure. The class of such objects subsumes what are called constructions in CxG, i.e. descriptions of form-meaning pairings, but will also include purely formal objects that have no meaning associated with them. Substantive idioms now pose no problem, since trees containing multiple words can be described in a single place, without privileging one of the words over the others.

By way of illustration, Figures 4 and 5 show (simplified) constructional LFG entries for *kicks the bucket* and *pulls strings*. A number of conventions are employed here. Firstly, for the sake of exposition, I use diagrams of c-structure and f-structure to stand in for the full list of constraints which describe these structures – for the

formal details, see Findlay 2019, ch. 5 and Findlay to appear, §7. Of course, this shorthand has some limitations. For example, *pull strings* can also be passivised, separated by *wh*-questioning and relativisation, etc., and these options are not represented by such a static diagram. There are different ways of permitting this flexibility. Most naturally, perhaps, the description of c-structure in the lexical entry can contain disjunctions over permitted elementary trees, organised in templates in an appropriate hierarchy.<sup>36</sup> This is equivalent to the standard approach in LTAG of using a METAGRAMMAR which captures generalisations across elementary trees (e.g. Crabbé *et al.* 2013). Another approach, that employed by Findlay (2019, 243–258), is to use lexical rules to map correspondences between different types of elementary tree. Whichever is used, we can constrain different idioms according to their different levels of flexibility, either by simply excluding the relevant structures from their descriptions, or by marking them so as to make them incompatible with the relevant lexical rules (Findlay 2019, 257–258).<sup>37</sup>

The second convention employed in Figures 4 and 5 is the use of simplified c-structures which follow X-bar theory even more loosely

---

<sup>36</sup>Note that the TAG approach to long-distance dependencies involves representing such dependencies locally, in an elementary tree; the filler and its gap can then be separated by adjunction of auxiliary trees between them. This means that the TREE FAMILY of a verb will include trees where its arguments are questioned, topicalised, relativised on, etc. See Abeillé and Rambow 2000 for an introduction to TAG, including the treatment of long-distance dependencies.

<sup>37</sup>It might be thought that explicitly describing such differences in syntactic flexibility in the grammar misses a generalisation: after all, as alluded to in the discussion of LA above, syntactic flexibility in idioms is supposed to correspond to semantic decomposability, as suggested by Nunberg *et al.* (1994) and assumed in much subsequent work. In fact, the empirical landscape is much more complex than this simple bifurcation would suggest. Fraser (1970), for instance, suggests a six-way classification of syntactic flexibility in idioms. In my own idiolect, it seems that some decomposable idioms are more flexible than others – for example, *pull strings* is far happier separated by relativisation or topicalisation than *spill the beans*. And Bargmann and Sailer (2018, 4, 20–21) present examples of non-decomposable idioms in German and English exhibiting syntactic flexibility. While there may ultimately be a semantic explanation for all of this, it does not seem unreasonable at present to allow for the grammar itself to have fine-grained control over syntactic flexibility.

than is common in LFG. This is both because many intervening bar-level nodes become unnecessary in a TAG as compared to a CFG, and in order to save space.

Figure 4:  
Constructional  
LFG entry for  
*kicks the bucket*

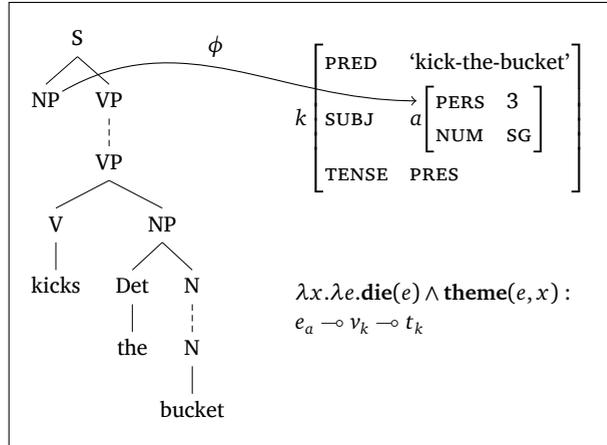
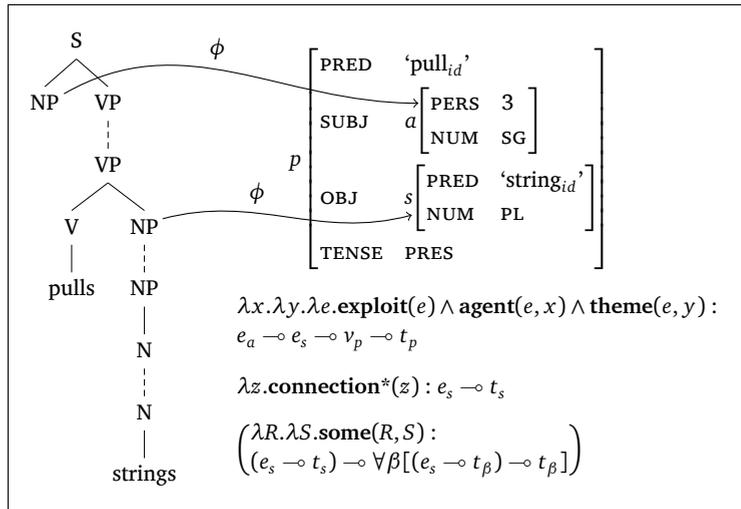


Figure 5:  
Constructional  
LFG entry for  
*pulls strings*



The third convention is the use of dashed lines in c-structures to represent simple dominance rather than immediate dominance: this enables adjunction at these nodes (see Vijay-Shanker 1992, 487–488 and Findlay 2019, 219–221), but if nothing is adjoined then the two nodes will be unified.

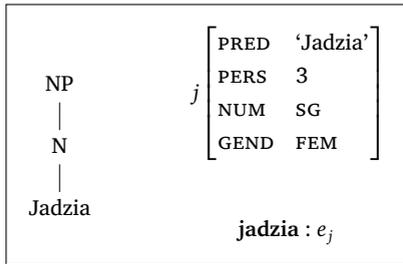


Figure 6:  
Constructional LFG entry for *Jadzia*

The fourth convention is that, to avoid clutter, I only show the  $\phi$  projection from maximal projections at c-structure to *embedded* f-structures – assume therefore that all undecorated maximal projections at c-structure map to the outermost f-structure shown, and that unannotated daughter nodes share the projection of their ancestors.

Finally, I only show c- and f-structures, along with the Glue Semantics meaning constructors, but of course full entries could also include information at other levels of representation within the projection architecture.

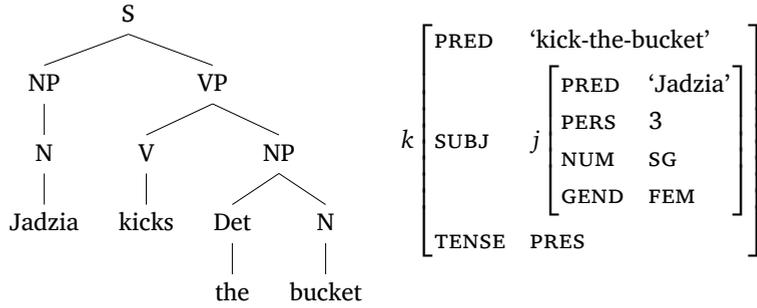
Turning now to the entries themselves, we see that the parallel representations of LFG allow us to illustrate what is the same and what is different across these two idioms. The fact that they both share the same surface form, that of a normal transitive VP, is shown by their c-structures, which are almost identical (the only difference is that *the bucket* already has its determiner fixed in the idiom, whereas the determiner position of *strings* is open).<sup>38</sup> The fact that *kick the bucket* cannot be decomposed but *pull strings* can is represented by (i) the latter having an articulated f-structure where the former does not, and (ii) by the latter contributing two different meaning constructors where the former only contributes one.

Both idioms take one external argument; for example, either of the descriptions in Figures 4 or 5 can combine with the entry for *Jadzia* in Figure 6, whose root node matches the open NP slot, and whose f-structure therefore unifies with the f-structure corresponding to that node. The resulting structures and Glue proof for *Jadzia kicks the bucket* are shown in Figure 7.

---

<sup>38</sup>This is exactly the kind of shared inheritance that would be captured in TAG by the metagrammar, and in constructional LFG by the template hierarchy.

Figure 7:  
Structures  
and Glue proof  
for *Jadzia kicks  
the bucket*



$$\frac{\lambda x. \lambda e. \mathbf{die}(e) \wedge \mathbf{theme}(e, x) : e_j \multimap v_k \multimap t_k \quad \mathbf{jadzia} : e_j}{\lambda e. \mathbf{die}(e) \wedge \mathbf{theme}(e, \mathbf{jadzia}) : v_k \multimap t_k \quad \lambda P. \exists e[P(e)] : (v_k \multimap t_k) \multimap t_k} \exists e[\mathbf{die}(e) \wedge \mathbf{theme}(e, \mathbf{jadzia}) : t_k]$$

Space precludes a full exploration of the possibilities of this new framework here, but see Findlay 2019, ch. 6 for a detailed demonstration of its application to a variety of substantive idioms, both verbal and non-verbal.<sup>39</sup> Hopefully what is clear is that by replacing the CFG standardly used to describe LFG’s c-structure with something more expressive, it is straightforward to fill in the gaps in LFG’s conception of EDL, extending it to every level of representation, and making LFG fully compatible with the assumptions of CxG.

<sup>39</sup>One area which necessitates further exploration is the challenge of non-configurational languages for a TAG-based c-structure. This is a particularly pressing concern for LFG, given that its treatment of non-configurationality is one of the parade examples of LFG’s utility as a framework. Although free word order languages go beyond the capabilities of TAG (Becker *et al.* 1991, 21–23), I am optimistic that, given the additional power of LFG, a suitably relaxed set of tree descriptions (e.g. removing statements of precedence relations from lexical entries so that order is underspecified in elementary trees) would be enough for constructional LFG to solve this problem. Nevertheless, this must be left for future work. Interestingly, some of the proposed extensions to TAG for tackling these problematic data (e.g. Multi-Component TAG: Weir 1988) are equivalent to LCFRSs (Kallmeyer 2010, 3), the same level of complexity attained by the tractable LFG of Wedekind and Kaplan (2020).

*Summary*

4.4

Because of the flexibility LFG permits in whether constraints are associated with lexical entries or with phrase-structure rules, the framework already has the ability to associate meaning with either words or phrases. It is therefore very capable of describing things like argument structure constructions and formal idioms. However, as idioms become more substantive, the framework begins to struggle. Although some substantive idioms can be treated lexically, as ‘words with spaces’, many cannot, and the most natural LFG solution, the lexical ambiguity approach, flies in the face of CxG dogma. The best solution, therefore, is to replace the unnecessarily restrictive CFG base of LFG with a more expressive TAG. By doing this, we give LFG the power to describe any kind of construction, formal or substantive, in a single place, just as is required of a CxG.

CONCLUSION

5

This paper has aimed to demonstrate that Lexical Functional Grammar (LFG) is a suitable framework for formalising Construction Grammar (CxG). I began by discussing some central assumptions of CxG, and then showed that LFG also subscribes to many of them. One area of disagreement is over the principle of Lexical Integrity (LI), which states that there is a strict separation between morphology and syntax. We saw that in fact both camps need to cede ground: LI in the strictest sense is too rigid, but a total abandonment of the morphology-syntax divide cannot be justified either.

Recent work in LFG (Asudeh *et al.* 2013) supports the contention that LFG is suitable for formalising CxG, in that it shows that the framework already has the capacity to handle many constructional phenomena. However, it turns out that this capacity is limited to formal idioms, and that substantive idioms are much more trouble. But if the context-free base of LFG is replaced with a more expressive formalism (in this case, a description-based TAG), LFG acquires the ability to describe arbitrarily large structures pairing phonological form with semantic content, enabling it to handle substantive idioms just as well

as formal ones. In this new version of LFG, the morphology-syntax divide is maintained, but the lexicon-grammar distinction is collapsed: since parsing just involves combining and satisfying stored collections of constraints, the lexicon, in a very real sense, *is* the grammar. Perhaps ironically, then, taking a more constructional view of things emphasises the *lexical* aspect of Lexical Functional Grammar.

## ACKNOWLEDGEMENTS

This work was carried out while I was employed under a Research Council of Norway grant (number 300495, “Universal Natural Language Understanding”), which I gratefully acknowledge. The seeds of this paper were sown several years earlier, however, during my doctoral research: this was funded by a UK Arts and Humanities Research Council studentship (grant reference AH/L503885/1), and owes much to my fantastic supervisors, Ash Asudeh and Mary Dalrymple. I would also like to thank the participants in the Workshop on Constructional Approaches in Formal Grammar at ICCG11 for their helpful and constructive comments and discussion, along with the three tremendously diligent JLM reviewers. Lastly, my thanks go to Laura Michaelis, Adam Przepiórkowski, and Manfred Sailer for their time and insights. Of course, acknowledgement here is no expectation of endorsement, and any errors in this work remain, as ever, my responsibility alone.

## REFERENCES

- Anne ABEILLÉ and Owen RAMBOW (2000), Tree Adjoining Grammar: an overview, in Anne ABEILLÉ and Owen RAMBOW, editors, *Tree Adjoining Grammars: formalisms, linguistic analysis and processing*, pp. 1–68, CSLI Publications, Stanford, CA.
- Peter ACKEMA and Ad NEELEMAN (2004), *Beyond morphology: interface conditions on word formation*, Oxford University Press, Oxford.
- Stephen R. ANDERSON (1992), *A-morphous morphology*, Cambridge University Press, Cambridge.

- Avery D. ANDREWS (2008), The role of PRED in LFG + Glue, in Miriam BUTT and Tracy Holloway KING, editors, *Proceedings of the LFG08 Conference*, pp. 46–67, CSLI Publications, Stanford, CA, <http://www.stanford.edu/group/cslipublications/cslipublications/LFG/13/papers/lfg08andrews.pdf>.
- Doug ARNOLD (2015), A Glue Semantics for structurally regular MWEs, poster presented at the PARSEME 5th general meeting, 23–24th September 2015, Iași, Romania.
- Ash ASUDEH (2012), *The logic of pronominal resumption*, Oxford University Press, Oxford.
- Ash ASUDEH, Mary DALRYMPLE, and Ida TOIVONEN (2013), Constructions with Lexical Integrity, *Journal of Language Modelling*, 1(1):1–54, <https://doi.org/10.15398/jlm.v1i1.56>.
- Ash ASUDEH and Gianluca GIORGOLO (2012), Flexible composition for optional and derived arguments, in Miriam BUTT and Tracy Holloway KING, editors, *Proceedings of the LFG12 Conference*, pp. 64–84, CSLI Publications, Stanford, CA, <http://www.stanford.edu/group/cslipublications/cslipublications/LFG/17/papers/lfg12asudehgiorgolo.pdf>.
- Ash ASUDEH, Gianluca GIORGOLO, and Ida TOIVONEN (2014), Meaning and valency, in Miriam BUTT and Tracy Holloway KING, editors, *Proceedings of the LFG14 Conference*, pp. 68–88, CSLI Publications, <http://web.stanford.edu/group/cslipublications/cslipublications/LFG/19/papers/lfg14asudehetal.pdf>.
- Ash ASUDEH, Paul B. MELCHIN, and Daniel SIDDIQI (2021), Constraints all the way down: DM in a representational model of grammar, in *Proceedings of the 39th West Coast Conference on Formal Linguistics*, Cascadilla Press, Somerville, MA.
- Ash ASUDEH and Daniel SIDDIQI (2022), Realizational morphosyntax in L<sub>R</sub>FG, in Miriam BUTT, Jamie Y. FINDLAY, and Ida TOIVONEN, editors, *Proceedings of the LFG'22 Conference*, pp. 21–40, CSLI Publications, Stanford, CA, <https://ojs.ub.uni-konstanz.de/lfg/index.php/main/article/view/20/>.
- Ash ASUDEH and Daniel SIDDIQI (to appear), Morphology in LFG, in Mary DALRYMPLE, editor, *The handbook of Lexical Functional Grammar*, Language Science Press, Berlin.
- Ash ASUDEH and Ida TOIVONEN (2015), Lexical-Functional Grammar, in Bernd HEINE and Heiko NARROG, editors, *The Oxford Handbook of Linguistic Analysis*, pp. 373–406, Oxford University Press, Oxford, second edition, <https://doi.org/10.1093/oxfordhb/9780199677078.013.0017>.
- Sascha BARGMANN and Manfred SAILER (2018), The syntactic flexibility of semantically non-decomposable idioms, in Manfred SAILER and Stella

MARKANTONATOU, editors, *Multiword expressions: insights from a multi-lingual perspective*, pp. 1–29, Language Science Press, Berlin, <http://langsci-press.org/catalog/book/184>.

Tilman BECKER, Aravind K. JOSHI, and Owen RAMBOW (1991), Long-distance scrambling and Tree Adjoining Grammars, in *Proceedings of the Fifth Conference of the European Chapter of the Association for Computational Linguistics (EACL 1991)*, Association for Computational Linguistics, Berlin, <https://aclanthology.org/E91-1005>.

Benjamin K. BERGEN and Nancy CHANG (2005), Embodied Construction Grammar in simulation-based language understanding, in Jan-Ola ÖSTMAN and Mirjam FRIED, editors, *Construction Grammars: cognitive grounding and theoretical extensions*, pp. 147–190, John Benjamins Publishing Co., Amsterdam.

Robert C. BERWICK (1982), Computational complexity and Lexical-Functional Grammar, *American Journal of Computational Linguistics*, 8(3–4):97–109, <https://dl.acm.org/citation.cfm?id=981926>.

Hans C. BOAS and Ivan A. SAG, editors (2012), *Sign-Based Construction Grammar*, number 193 in CSLI Lecture Notes, CSLI Publications, Stanford, CA.

J. Kathryn BOCK and Willem J. M. LEVELT (1994), Language production: grammatical encoding, in Morton A. GERNSBACHER, editor, *Handbook of psycholinguistics*, pp. 945–984, Academic Press, San Diego, CA.

Rens BOD and Ronald KAPLAN (1998), A probabilistic corpus-driven model for Lexical-Functional analysis, in *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics, Volume 1*, pp. 145–151, Association for Computational Linguistics, <http://www.aclweb.org/anthology/P98-1022>.

Dwight L. BOLINGER (1968), Entailment and the meaning of structures, *Glossa*, 2(2):119–127.

Kersti BÖRJARS, Rachel NORDLINGER, and Louisa SADLER (2019), *Lexical-Functional Grammar: an introduction*, Cambridge University Press, Cambridge, doi:10.1017/9781316756584.

Joan BRESNAN (1982), The passive in lexical theory, in Joan BRESNAN, editor, *The mental representation of grammatical relations*, pp. 3–86, MIT Press, Cambridge, MA.

Joan BRESNAN (1995), Linear order, syntactic rank, and empty categories: on weak crossover, in Mary DALRYMPLE, Ronald M. KAPLAN, John T. MAXWELL, III, and Annie ZAENEN, editors, *Formal issues in Lexical-Functional Grammar*, pp. 241–274, CSLI Publications, Stanford, CA.

Joan BRESNAN (1998), Morphology competes with syntax: explaining typological variation in weak crossover effects, in Pilar BARBOSA, Danny FOX,

Paul HAGSTROM, Martha MCGINNIS, and David PESETSKY, editors, *Is the best good enough? Optimality and competition in syntax*, MIT Press and MIT Working Papers in Linguistics, Cambridge, MA.

Joan BRESNAN, Ash ASUDEH, Ida TOIVONEN, and Stephen WECHSLER (2016), *Lexical-functional syntax*, Wiley-Blackwell, Oxford, second edition.

Joan BRESNAN and Sam A. MCHOMBO (1987), Topic, pronoun, and agreement in Chicheŵa, *Language*, 63(4):741–782.

Joan BRESNAN and Sam A. MCHOMBO (1995), The lexical integrity principle: evidence from Bantu, *Natural Language and Linguistic Theory*, 13(2):181–254, <https://doi.org/10.1007/BF00992782>.

Benjamin BRUENING (2018), The lexicalist hypothesis: both wrong and superfluous, *Language*, 94(1):1–42, <https://doi.org/10.1353/lan.2018.0000>.

Miriam BUTT, Mary DALRYMPLE, and Anette FRANK (1997), An architecture for linking theory in LFG, in Miriam BUTT and Tracy Holloway KING, editors, *Proceedings of the LFG97 Conference*, CSLI Publications, Stanford, CA, <http://web.stanford.edu/group/cslipublications/cslipublications/LFG/LFG2-1997/lfg97butt-dalrymple-frank.pdf>.

Miriam BUTT, Helge DYVIK, Tracy Holloway KING, Hiroshi MASUICHI, and Christian ROHRER (2002), The Parallel Grammar project, in *Proceedings of the workshop on grammar engineering and evaluation (COLING-02)*, Association for Computational Linguistics, <https://www.aclweb.org/anthology/W02-1503.pdf>.

Miriam BUTT, María-Eugenia NIÑO, and Frédérique SEGOND (1996), Multilingual processing of auxiliaries in LFG, in Dafydd GIBBON, editor, *Natural language processing and speech technology: results of the 3rd KONVENS conference, Bielefeld, October 1996*, pp. 111–122.

Bert CAPPELLE (2022), Lexical Integrity: a mere construct or more a construction?, *Yearbook of the German Cognitive Linguistics Association*, 10:183–216, <https://doi.org/10.1515/gcla-2022-0009>.

Andrew CARNIE (2000), On the definition of  $X^0$  and XP, *Syntax*, 3(2):59–106.

Noam CHOMSKY (1957), *Syntactic structures*, Mouton, The Hague.

Noam CHOMSKY (1970), Remarks on nominalization, in Roderick A. JACOBS and Peter S. ROSENBAUM, editors, *Readings in English transformational grammar*, pp. 184–221, Ginn, Waltham, MA.

Noam CHOMSKY (1993), A minimalist program for linguistic theory, in Kenneth HALE and Samuel KEYSER, editors, *The view from Building 20: essays in linguistics in honor of Sylvain Bromberger*, pp. 1–52, MIT Press, Cambridge, MA.

Noam CHOMSKY (1995), *The Minimalist Program*, Cambridge University Press, Cambridge.

Benoît CRABBÉ, Denys DUCHIER, Claire GARDENT, Joseph Le ROUX, and Yannick PARMENTIER (2013), XMG: eXtensible MetaGrammar, *Computational Linguistics*, 39(3):591–629, doi:10.1162/COLI\_a\_00144.

William CROFT (2001), *Radical Construction Grammar: syntactic theory in typological perspective*, Oxford University Press, Oxford, doi:10.1093/acprof:oso/9780198299554.001.0001.

Dick CROUCH, Mary DALRYMPLE, Ronald M. KAPLAN, Tracy Holloway KING, John T. MAXWELL III, and Paula NEWMAN (2017), *XLE documentation*, Palo Alto Research Center (PARC), Palo Alto, CA., [https://ling.sprachwiss.uni-konstanz.de/pages/xle/doc/xle\\_toc.html](https://ling.sprachwiss.uni-konstanz.de/pages/xle/doc/xle_toc.html).

Peter W. CULICOVER and Ray JACKENDOFF (1999), The view from the periphery: the English comparative correlative, *Linguistic Inquiry*, 30(4):543–571.

Peter W. CULICOVER and Ray JACKENDOFF (2005), *Simpler syntax*, Oxford University Press, Oxford.

Chris CULY (1985), The complexity of the vocabulary of Bambara, *Linguistics and Philosophy*, 8(3):345–351, doi:10.1007/978-94-009-3401-6\_14.

Haskell B. CURRY and Robert FEYS (1958), *Combinatory logic: volume I*, North Holland, Amsterdam.

Mary DALRYMPLE, editor (1999), *Semantics and syntax in Lexical Functional Grammar: the resource logic approach*, MIT Press, Cambridge, MA.

Mary DALRYMPLE (2001), *Lexical Functional Grammar*, number 34 in *Syntax and Semantics*, Academic Press, San Diego, CA.

Mary DALRYMPLE (2015), Morphology in the LFG architecture, in Miriam BUTT and Tracy Holloway KING, editors, *Proceedings of the LFG15 Conference*, pp. 64–83, CSLI Publications, Stanford, CA, <http://web.stanford.edu/group/cslipublications/cslipublications/LFG/20/papers/lfg15dalrymple.pdf>.

Mary DALRYMPLE, editor (forthcoming), *The Handbook of Lexical Functional Grammar*, Empirically Oriented Theoretical Morphology and Syntax, Language Science Press, Berlin.

Mary DALRYMPLE and Jamie Y. FINDLAY (2019), Lexical Functional Grammar, in András KERTÉSZ, Edith MORAVCSIK, and Csilla RÁKOSI, editors, *Current approaches to syntax: a comparative handbook*, pp. 123–154, De Gruyter Mouton, Berlin, doi:10.1515/9783110540253-005.

Mary DALRYMPLE, Ronald M. KAPLAN, and Tracy Holloway KING (2001), Weak crossover and the absence of traces, in Miriam BUTT and Tracy Holloway

KING, editors, *Proceedings of the LFG01 Conference*, pp. 66–82, CSLI Publications, Stanford, CA, <https://typo.uni-konstanz.de/lfg-proceedings/LFGprocCSLI/LFG2001/pdfs/lfg01dalrympleetal.pdf>.

Mary DALRYMPLE, Ronald M. KAPLAN, and Tracy Holloway KING (2004), Linguistic generalizations over descriptions, in Miriam BUTT and Tracy Holloway KING, editors, *Proceedings of the LFG04 Conference*, pp. 199–208, CSLI Publications, Stanford, CA, <https://web.stanford.edu/group/cslipublications/cslipublications/LFG/9/pdfs/lfg04dkk.pdf>.

Mary DALRYMPLE, Ronald M. KAPLAN, and Tracy Holloway KING (2007), The absence of traces: evidence from weak crossover, in Annie ZAENEN, Jane SIMPSON, Tracy Holloway KING, Jane GRIMSHAW, Joan MALING, and Chris MANNING, editors, *Architectures, rules, and preferences: variations on themes by Joan W. Bresnan*, pp. 85–102, CSLI Publications, Stanford, CA.

Mary DALRYMPLE, Ronald M. KAPLAN, and Tracy Holloway KING (2015), Economy of expression as a principle of syntax, *Journal of Language Modelling*, 2(3):377–412, doi:10.15398/jlm.v3i2.82.

Mary DALRYMPLE and Tracy Holloway KING (2013), Nested and crossed dependencies and the existence of traces, in Tracy Holloway KING and Valeria DE PAIVA, editors, *From quirky case to representing space: papers in honor of Annie Zaenen*, pp. 139–151, CSLI Publications, Stanford, CA.

Mary DALRYMPLE, John LAMPING, and Vijay SARASWAT (1993), LFG semantics via constraints, in Steven KRAUWER, Michael MOORTGAT, and Louis DES TOMBE, editors, *Proceedings of the Sixth Conference of the European Chapter of the Association for Computational Linguistics (EACL 1993)*, pp. 97–105, <https://www.aclweb.org/anthology/E93-1013.pdf>.

Mary DALRYMPLE, John J. LOWE, and Louise MYCOCK (2019), *The Oxford reference guide to Lexical Functional Grammar*, Oxford University Press, Oxford, doi:10.1093/oso/9780198733300.001.0001.

Mary DALRYMPLE and Louise MYCOCK (2011), The prosody-semantics interface, in Miriam BUTT and Tracy Holloway KING, editors, *Proceedings of the LFG11 Conference*, pp. 173–193, CSLI Publications, Stanford, CA, <http://web.stanford.edu/group/cslipublications/cslipublications/LFG/16/papers/lfg11dalrymplemycock.pdf>.

Mary DALRYMPLE and Irina NIKOLAEVA (2011), *Objects and information structure*, Cambridge University Press, Cambridge, doi:10.1017/CBO9780511993473.

Jared DESJARDINS (2023), A cross-theoretical and cross-linguistic survey of lexical integrity and the nature of the morphology–syntax interface, unpublished manuscript, University of Colorado at Boulder, <https://jared-desjardins.github.io/LI/docs/home/>.

Anna Maria DI SCIULLO and Edwin WILLIAMS (1987), *On the definition of word*, number 14 in *Linguistic Inquiry Monographs*, MIT Press, Cambridge, MA.

Helge DYVIK, Gyri Smørdal LOSNEGAARD, and Victoria ROSÉN (2019), Multiword expressions in an LFG grammar for Norwegian, in Yannick PARMENTIER and Jakub WASZCZUK, editors, *Representation and parsing of multiword expressions: current trends*, pp. 69–108, Language Science Press, Berlin, doi:10.5281/zenodo.2579037.

Yehuda N. FALK (2001), *Lexical-Functional Grammar: an introduction to parallel constraint-based syntax*, CSLI Publications, Stanford, CA.

Yehuda N. FALK (2010), An unmediated analysis of relative clauses, in Miriam BUTT and Tracy Holloway KING, editors, *Proceedings of the LFG10 Conference*, pp. 207–227, CSLI Publications, <http://web.stanford.edu/group/cslipublications/cslipublications/LFG/15/papers/lfg10falk.pdf>.

Charles J. FILLMORE (1985), Syntactic intrusions and the notion of grammatical constructions, in Mary NIEPOKUJ, Mary VANCLAY, Vassiliki NIKIFORIDOU, and Deborah FEDER, editors, *Proceedings of the Eleventh Annual Meeting of the Berkeley Linguistics Society*, pp. 73–86, Berkeley Linguistic Society, Berkeley, CA.

Charles J. FILLMORE (1987), Varieties of conditional sentences, in Fred MARSHALL, editor, *Proceedings of the Third Eastern States Conference on Linguistics (ESCOL 86): October 10–11, 1986*, pp. 163–182, University of Pittsburgh, Pittsburgh, PA.

Charles J. FILLMORE (1988), The mechanisms of ‘Construction Grammar’, in Shelley AXMAKER, Annie JAISSE, and Helen SINGMASTER, editors, *Proceedings of the Fourteenth Annual Meeting of the Berkeley Linguistics Society*, pp. 35–55, Berkeley Linguistics Society, University of California, Berkeley, Berkeley, CA.

Charles J. FILLMORE, Paul KAY, and Mary Catherine O’CONNOR (1988), Regularity and idiomacity in grammatical constructions: the case of *let alone*, *Language*, 64(3):501–538, doi:10.2307/414531.

Jamie Y. FINDLAY (2016), Mapping theory without argument structure, *Journal of Language Modelling*, 4(2):293–338, doi:10.15398/jlm.v4i2.171.

Jamie Y. FINDLAY (2017), Multiword expressions and lexicalism, in Miriam BUTT and Tracy Holloway KING, editors, *Proceedings of the LFG17 Conference*, pp. 209–229, CSLI Publications, Stanford, CA, <http://web.stanford.edu/group/cslipublications/cslipublications/LFG/LFG-2017/lfg2017-findlay.pdf>.

Jamie Y. FINDLAY (2019), *Multiword expressions and the lexicon*, D.Phil. thesis, University of Oxford, <https://ora.ox.ac.uk/objects/uuid:502e8ca5-02f7-4be4-8778-cd89364ba670>.

Jamie Y. FINDLAY (2020), Mapping Theory and the anatomy of a lexical entry, in Miriam BUTT and Ida TOIVONEN, editors, *Proceedings of the LFG20*

Conference, pp. 127–147, CSLI Publications, Stanford, CA,  
[http://web.stanford.edu/group/cslipublications/  
cslipublications/LFG/LFG-2020/lfg2020-findlay.pdf](http://web.stanford.edu/group/cslipublications/cslipublications/LFG/LFG-2020/lfg2020-findlay.pdf).

Jamie Y. FINDLAY (2021), Meaning in LFG, in I. Wayan ARKA, Ash ASUDEH, and Tracy Holloway KING, editors, *Modular design of grammar: linguistics on the edge*, pp. 340–374, Oxford University Press,  
doi:10.1093/oso/9780192844842.003.0020.

Jamie Y. FINDLAY (to appear), LFG and Tree-Adjoining Grammar, in Mary DALRYMPLE, editor, *The handbook of Lexical Functional Grammar*, Language Science Press, Berlin.

Ingrid FISCHER and Martina KEIL (1996), Parsing decomposable idioms, in Jun-ichi TSUJII, editor, *COLING 1996 Volume 1: The 16th International Conference on Computational Linguistics*, pp. 388–393, Association for Computational Linguistics, Copenhagen,  
<https://aclanthology.org/C96-1066>.

Anette FRANK and Annie ZAENEN (2004), Tense in LFG: syntax and morphology, in Louisa SADLER and Andrew SPENCER, editors, *Projecting morphology*, pp. 23–66, CSLI Publications, Stanford, CA.

Bruce FRASER (1970), Idioms within a transformational grammar, *Foundations of Language*, 6(1):22–42, <https://www.jstor.org/stable/25000426>.

Gerald GAZDAR, Ewan KLEIN, Geoffrey K. PULLUM, and Ivan A. SAG (1985), *Generalized Phrase Structure Grammar*, Harvard University Press, Cambridge, MA.

Jila GHOMESHI, Ray JACKENDOFF, Nicole ROSEN, and Kevin RUSSELL (2004), Contrastive focus reduplication in English (the salad-salad paper), *Natural Language & Linguistic Theory*, 22(2):307–357,  
doi:10.1023/B:NALA.0000015789.98638.f9.

Jean-Yves GIRARD (1987), Linear logic, *Theoretical Computer Science*, 50(1):1–102, doi:10.1016/0304-3975(87)90045-4.

Adele E. GOLDBERG (1995), *Constructions: a Construction Grammar approach to argument structure*, University of Chicago Press, Chicago, IL.

Adele E. GOLDBERG (2006), *Constructions at work: the nature of generalization in language*, Oxford University Press, Oxford.

Adele E. GOLDBERG (2013), Argument structure constructions versus lexical rules or derivational verb templates, *Mind & Language*, 28(4),  
doi:10.1111/mila.12026.

Adele E. GOLDBERG and Ray JACKENDOFF (2004), The English resultative as a family of constructions, *Language*, 80(3):532–568,  
<https://www.jstor.org/stable/4489722>.

Morris HALLE and Alec MARANTZ (1993), Distributed Morphology and the pieces of inflection, in Kenneth HALE and Samuel Jay KEYSER, editors, *The view from Building 20: Essays in linguistics in honor of Sylvain Bromberger*, pp. 111–176, MIT Press, Cambridge, MA.

William A. HOWARD (1980), The formulae-as-types notion of construction, in *To H. B. Curry: essays on combinatory logic, lambda calculus, and formalism*, pp. 479–490, Academic Press, London, circulated in unpublished form from 1969.

Ray JACKEDOFF (2008), *Construction after construction* and its theoretical challenges, *Language*, 84(1):8–28, doi:10.1353/lan.2008.0058.

Ray JACKENDOFF (1977),  $\bar{X}$  syntax: a study of phrase structure, number 2 in *Linguistic Inquiry Monographs*, MIT Press, Cambridge, MA.

Ray JACKENDOFF (1992), Babe Ruth homered his way into the hearts of America, in Tim STOWELL and Eric WEHRLI, editors, *Syntax and the lexicon*, number 26 in *Syntax and Semantics*, pp. 155–178, Academic Press, San Diego, CA.

Aravind K. JOSHI (2005), Tree-adjointing grammars, in Ruslan MITKOV, editor, *The Oxford handbook of computational linguistics*, pp. 483–498, Oxford University Press, Oxford, first edition, doi:10.1093/oxfordhb/9780199276349.013.0026.

Aravind K. JOSHI, Leon S. LEVY, and Masako TAKAHASHI (1975), Tree adjunct grammars, *Journal of Computer and System Sciences*, 10(1):136–163, doi:10.1016/S0022-0000(75)80019-5.

Aravind K. JOSHI and Yves SCHABES (1997), Tree-Adjoining Grammars, in Grzegorz ROZENBERG and Arto SALOMAA, editors, *Handbook of formal languages, volume 3: beyond words*, pp. 69–123, Springer, Berlin.

Daniel JURAFSKY (1992), *An on-line computational model of human sentence interpretation: a theory of the representation and use of linguistic knowledge*, Ph.D. thesis, University of California at Berkeley.

Laura KALLMEYER (2010), *Parsing beyond context-free grammars*, Springer, Heidelberg.

Ronald M. KAPLAN and Joan BRESNAN (1982), Lexical-Functional Grammar: a formal system for grammatical representation, in Joan BRESNAN, editor, *The mental representation of grammatical relations*, pp. 173–281, MIT Press, Cambridge, MA.

Ronald M. KAPLAN, John T. MAXWELL, III, and Annie ZAENEN (1987), Functional uncertainty, in *CSLI Publications Monthly Newsletter*, Stanford University, Stanford, CA.

Ronald M. KAPLAN and Paula S. NEWMAN (1997), Lexical resource reconciliation in the Xerox Linguistic Environment, in *Proceedings of the ACL*

*Workshop on Computational Environments for Grammar Development and Engineering*, Association for Computational Linguistics.

Paul KAY and Charles J. FILLMORE (1999), Grammatical constructions and linguistic generalizations: the *What's X doing Y?* construction, *Language*, 75(1):1–33, doi:10.2307/417472.

Paul KAY, Ivan A. SAG, and Daniel P. FLICKINGER (2015), A lexical theory of phrasal idioms, unpublished manuscript, CSLI, Stanford: <http://www1.icsi.berkeley.edu/~kay/idiom-pdf/latex.11-13-15.pdf>.

Anna KIBORT (2007), Extending the applicability of Lexical Mapping Theory, in Miriam BUTT and Tracy Holloway KING, editors, *Proceedings of the LFG07 Conference*, pp. 250–270, CSLI Publications, Stanford, CA, <http://www.stanford.edu/group/cslipublications/cslipublications/LFG/12/papers/lfg07kibort.pdf>.

Tracy Holloway KING (1995), *Configuring topic and focus in Russian*, (Stanford University Dissertations in Linguistics), CSLI Publications, Stanford, CA, revised and corrected version of 1993 Stanford University dissertation.

Miltiadis KOKKONIDIS (2008), First-order Glue, *Journal of Logic, Language and Information*, 17(1):43–68, doi:10.1007/s10849-006-9031-0.

George LAKOFF (1987), *Women, fire, and dangerous things: what categories reveal about the mind*, University of Chicago Press, Chicago, IL.

Stephen Guy LAPOINTE (1980), *A theory of grammatical agreement*, Ph.D. thesis, University of Massachusetts at Amherst.

Timm LICHTÉ and Laura KALLMEYER (2016), Same syntax, different semantics: a compositional approach to idiomaticity in multi-word expressions, in Christopher PIÑÓN, editor, *Empirical Issues in Syntax and Semantics 11*, pp. 111–140, Colloque de Syntaxe et Sémantique à Paris (CSSP), Paris, [http://www.cssp.cnrs.fr/eiss11/eiss11\\_lichte-and-kallmeyer.pdf](http://www.cssp.cnrs.fr/eiss11/eiss11_lichte-and-kallmeyer.pdf).

Timm LICHTÉ and Laura KALLMEYER (2017), Tree-Adjoining Grammar: a tree-based constructionist grammar framework for natural language understanding, in *Proceedings of the AAAI 2017 Spring Symposium on Computational Construction Grammar and Natural Language Understanding*, Technical Report SS-17-02, pp. 205–212, Association for the Advancement of Artificial Intelligence, <https://www.aaai.org/ocs/index.php/SSS/SSS17/paper/viewFile/15330/14536>.

Rochelle LIEBER and Sergio SCALISE (2007), The Lexical Integrity Hypothesis in a new theoretical universe, in Geert BOOIJ, Luca DUCCESCHI, Bernard FRADIN, Emiliano GUEVARA, Angela RALLI, and Sergio SCALISE, editors, *On-line proceedings of the fifth Mediterranean Morphology Meeting (MMM5)*, pp. 1–24, Università degli Studi di Bologna, Bologna.

John J. LOWE (2014), Gluing meanings and semantic structures, in Miriam BUTT and Tracy Holloway KING, editors, *Proceedings of the LFG14 Conference*, pp. 387–407, CSLI Publications, Stanford, CA, <http://web.stanford.edu/group/cslipublications/cslipublications/LFG/19/papers/lfg14lowe.pdf>.

John J. LOWE (2015), The syntax of Sanskrit compounds, *Language*, 91(3):e71–e115, <https://doi.org/10.1353/lan.2015.0034>.

Alec MARANTZ (1997), No escape from syntax: don't try morphological analysis in the privacy of your own lexicon, in Alexis DIMITRIADIS, Laura SIEGEL, Clarissa SUREK-CLARK, and Alexander WILLIAMS, editors, *University of Pennsylvania working papers in linguistics, volume 4.2: Proceedings of the 21st Annual Penn Linguistics Colloquium*, pp. 201–225, University of Pennsylvania Department of Linguistics, Philadelphia, PA.

Jean-Philippe MARCOTTE (2009), Anti-extraction in Québécois French wh-interrogatives, unpublished manuscript, University of Minnesota.

Jean-Philippe MARCOTTE and Kateryna KENT (2010), Russian verbal affixes in the projection architecture, in Miriam BUTT and Tracy Holloway KING, editors, *Proceedings of the LFG10 Conference*, pp. 353–373, CSLI Publications, Stanford, CA, <https://web.stanford.edu/group/cslipublications/cslipublications/LFG/15/papers/lfg10marcottekent.pdf>.

Matthew S. MCGLONE, Sam GLUCKSBERG, and Cristina CACCIARI (1994), Semantic productivity and idiom comprehension, *Discourse Processes*, 17(2):167–190, <https://doi.org/10.1080/01638539409544865>.

Paul B. MELCHIN, Ash ASUDEH, and Dan SIDDIQI (2020), Ojibwe agreement in Lexical-Realizational Functional Grammar, in Miriam BUTT and Ida TOIVONEN, editors, *Proceedings of the LFG'20 Conference, On-Line*, pp. 268–288, CSLI Publications, Stanford, CA, <http://cslipublications.stanford.edu/LFG/2020/lfg2020-mas.pdf>.

Laura A. MICHAELIS (2015), Sign-Based Construction Grammar, in Bernd HEINE and Heiko NARROG, editors, *The Oxford handbook of linguistic analysis* (2nd edn.), pp. 147–166, Oxford University Press, Oxford, doi:10.1093/oxfordhb/9780199677078.013.0007.

Stefan MÜLLER (2002), Syntax or morphology: German particle verbs revisited, in Nicole DEHÉ, Ray JACKENDOFF, Andrew MCINTYRE, and Silke URBAN, editors, *Verb particle explorations*, number 1 in Interface exploration, pp. 119–139, Mouton de Gruyter, Berlin, doi:10.1515/9783110902341.119.

Stefan MÜLLER (2006), Phrasal or lexical constructions?, *Language*, 82(4):850–883, doi:10.1353/lan.2006.0213.

Stefan MÜLLER (2018), *A lexicalist account of argument structure: template-based phrasal LFG approaches and a lexical HPSG alternative*, number 2 in Conceptual

Foundations of Language Science, Language Science Press, Berlin,  
doi:10.5281/zenodo.1441351.

Stefan MÜLLER (2021), HPSG and Construction Grammar, in Stefan MÜLLER, Anne ABEILLÉ, Robert D. BORSLEY, and Jean-Pierre KOENIG, editors, *Head-Driven Phrase Structure Grammar: the handbook*, number 9 in Empirically Oriented Theoretical Morphology and Syntax, pp. 1497–1553, Language Science Press, Berlin, doi:10.5281/zenodo.5599882.

Stefan MÜLLER and Stephen WECHSLER (2014), Lexical approaches to argument structure, *Theoretical Linguistics*, 40(1–2):1–76,  
<https://doi.org/10.1515/tl-2014-0001>.

Louise MYCOCK and John LOWE (2013), The prosodic marking of discourse functions, in Miriam BUTT and Tracy Holloway KING, editors, *Proceedings of the LFG13 Conference*, pp. 440–460, CSLI Publications, Stanford, CA,  
<http://web.stanford.edu/group/cslipublications/cslipublications/LFG/18/papers/lfg13mycocklowe.pdf>.

Perna NADATHUR (2013), Weak crossover and the Direct Association Hypothesis, in Miriam BUTT and Tracy Holloway KING, editors, *Proceedings of the LFG13 Conference*, pp. 461–481, CSLI Publications, Stanford, CA,  
<http://web.stanford.edu/group/cslipublications/cslipublications/LFG/18/papers/lfg13nadathur.pdf>.

Ryuichi NAKANISHI, Hiroyuki SEKI, and Tadao KASAMI (1992), On the generative capacity of lexical-functional grammars, *IEICE Transactions on Information and Systems*, E75-D(4):509–516,  
[https://search.ieice.org/bin/summary.php?id=e75-d\\_4\\_509](https://search.ieice.org/bin/summary.php?id=e75-d_4_509).

Rachel NORDLINGER (1998), *Constructive case*, CSLI Publications. Revised version of Stanford University PhD Thesis, Stanford, CA.

Geoffrey NUNBERG, Ivan A. SAG, and Thomas WASOW (1994), Idioms, *Language*, 70(3):491–538, doi:10.1353/lan.1994.0007.

Terence PARSONS (1990), *Events in the semantics of English: a study in subatomic semantics*, MIT Press, Cambridge, MA.

Barbara PARTEE, Alice TER MEULEN, and Robert E. WALL (1990), *Mathematical methods in linguistics*, Kluwer, Dordrecht.

Adam PRZEPIÓRKOWSKI (2017), A full-fledged hierarchical lexicon in LFG: the FrameNet approach, in Victoria ROSÉN and Koenraad De SMEDT, editors, *The very model of a modern linguist: in honor of Helge Dyvik*, number 8 in Bergen Language and Linguistics Studies, pp. 202–219, University of Bergen, Bergen, doi:10.15845/bells.v8i1.1336.

Adam PRZEPIÓRKOWSKI and Anna KUPŚĆ (2006), HPSG for Slavicists, *Glossos*, 8, <https://slaviccenters.duke.edu/projects/glossos-journal/issues/issue-8>.

Geoffrey K. PULLUM and Kyle RAWLINS (2007), Argument or no argument?, *Linguistics and Philosophy*, 30(2):277–287, doi:10.1007/s10988-007-9013-y.

Brian ROARK and Richard SPROAT (2007), *Computational approaches to morphology and syntax*, Oxford Surveys in Syntax and Morphology, Oxford University Press, Oxford.

Ivan A. SAG (1997), English relative clause constructions, *Journal of Linguistics*, 33(2):431–484, <https://www.jstor.org/stable/4176423>.

Ivan A. SAG (2007), Remarks on locality, in Stefan MÜLLER, editor, *Proceedings of the 14th International Conference on Head-Driven Phrase Structure Grammar*, pp. 394–414, CSLI Publications, Stanford, CA, doi:10.21248/hpsg.2007.23.

Ivan A. SAG (2010), English filler-gap constructions, *Language*, 86(3):486–545.

Ivan A. SAG (2012), Sign-Based Construction Grammar: an informal synopsis, in Hans C. BOAS and Ivan A. SAG, editors, *Sign-Based Construction Grammar*, number 193 in CSLI Lecture Notes, pp. 69–202, CSLI Publications, Stanford, CA.

Ivan A. SAG, Timothy BALDWIN, Francis BOND, Ann COPESTAKE, and Dan FLICKINGER (2002), Multiword expressions: a pain in the neck for NLP, in Alexander GELBUKH, editor, *Computational Linguistics and Intelligent Text Processing: Third International Conference (CICLing 2002)*, number 2276 in Lecture Notes in Computer Science, pp. 1–15, Springer, Berlin, doi:10.1007/3-540-45715-1\_1.

Manfred SAILER (2000), *Combinatorial semantics and idiomatic expressions in Head-Driven Phrase Structure Grammar*, Ph.D. thesis, Eberhard-Karls-Universität Tübingen.

Yves SCHABES, Anne ABEILLÉ, and Aravind K. JOSHI (1988), Parsing strategies with ‘lexicalized’ grammars: application to Tree Adjoining Grammars, in *Proceedings of the 12th Conference on Computational Linguistics (COLING)*, pp. 578–583, Association for Computational Linguistics, doi:10.3115/991719.991757.

Stuart M. SHIEBER (1985), Evidence against the context-freeness of natural language, *Linguistics and Philosophy*, 8(3):333–343, doi:10.1007/978-94-009-3401-6\_12.

Dorothy C. SIEGEL (1974), *Topics in English morphology*, Ph.D. thesis, MIT.

Jane SIMPSON (1983), *Aspects of Warlpiri morphology and syntax*, Ph.D. thesis, MIT.

Jane SIMPSON (1991), *Warlpiri morpho-syntax: a lexicalist approach*, Kluwer Academic Publishers, Dordrecht.

Liselotte SNIJDERS (2012), Issues concerning constraints on discontinuous NPs in Latin, in Miriam BUTT and Tracy Holloway KING, editors, *Proceedings of the LFG12 Conference*, pp. 565–581, CSLI Publications, Stanford, CA,

<http://www.stanford.edu/group/cslipublications/cslipublications/LFG/17/papers/lfg12snijders.pdf>.

Liselotte SNIJDERS (2015), *The nature of configurationality in LFG*, Ph.D. thesis, University of Oxford, <https://ora.ox.ac.uk/objects/uuid:1310f160-283e-411e-a8d7-20ab4b3380c2>.

Andrew SPENCER (2005a), Case in Hindi, in Miriam BUTT and Tracy Holloway KING, editors, *Proceedings of the LFG05 Conference*, pp. 429–446, CSLI Publications, Stanford, CA, <https://web.stanford.edu/group/cslipublications/cslipublications/LFG/10/pdfs/lfg05spencer.pdf>.

Andrew SPENCER (2005b), Word formation and syntax, in Pavol ŠTEKAUER and Rochelle LIEBER, editors, *Handbook of word-formation*, number 64 in Studies in Natural Language and Linguistic Theory, pp. 73–97, Springer, Dordrecht.

Luc STEELS, editor (2011), *Design patterns in Fluid Construction Grammar*, number 11 in Constructional Approaches to Language, John Benjamins Publishing Co., Amsterdam, doi:10.1075/cal.11.

Luc STEELS and Remi VAN TRIJP (2011), How to make Construction Grammars fluid and robust, in Luc STEELS, editor, *Design patterns in Fluid Construction Grammar*, pp. 301–330, John Benjamins, Amsterdam.

Steven L. STRAUSS (1982), *Lexicalist phonology of English and German*, Foris, Dordrecht.

Ida TOIVONEN (2002), The directed motion construction in Swedish, *Journal of Linguistics*, 38(2):313–345, doi:10.1017/S002222670200141X.

Ida TOIVONEN (2003), *Non-projecting words: a case study of Swedish particles*, Kluwer Academic Publishers, Dordrecht.

Michael TOMASELLO (2003), *Constructing a language: a usage-based theory of language acquisition*, Harvard University Press, Cambridge, MA.

K. VIJAY-SHANKER (1992), Using descriptions of trees in a Tree Adjoining Grammar, *Computational Linguistics*, 18(4):481–517, <https://dl.acm.org/citation.cfm?id=176317>.

Yang WANG and Tim HUNTER (2023), On regular copying languages, *Journal of Language Modelling*, 11(1):1–66, doi:10.15398/jlm.v11i1.342, <https://jlm.ipipan.waw.pl/index.php/JLM/article/view/342>.

Jürgen WEDEKIND and Ronald M. KAPLAN (2020), Tractable Lexical-Functional Grammar, *Computational Linguistics*, 46(3):515–569, doi:10.1162/coli\_a\_00384.

David WEIR (1988), *Characterizing mildly context-sensitive grammar formalisms*, Ph.D. thesis, University of Pennsylvania.

*Jamie Y. Findlay*

Edwin WILLIAMS (2007), Dumping lexicalism, in Gillian RAMCHAND and Charles REISS, editors, *The Oxford handbook of linguistic interfaces*, pp. 353–382, Oxford University Press, Oxford,  
doi:10.1093/oxfordhb/9780199247455.013.0012.

Annie ZAENEN (1989), Nominal arguments in Dutch and WYSIWYG LFG, unpublished manuscript, XEROX PARC.

*Jamie Y. Findlay*

© 0000-0002-7700-1525

jamie.findlay@iln.uio.no

Department of Linguistics  
and Nordic Studies  
University of Oslo  
Postboks 1102  
Blindern  
0317 OSLO

Jamie Y. Findlay (2023), *Lexical Functional Grammar as a Construction Grammar*, *Journal of Language Modelling*, 11(2):197–266

doi <https://dx.doi.org/10.15398/jlm.v11i2.338>

This work is licensed under the *Creative Commons Attribution 4.0 Public License*.

cc  <http://creativecommons.org/licenses/by/4.0/>