Constructions with Lexical Integrity: Templates as the Lexicon–Syntax Interface

LFG differs from Construction Grammar (CG) in assuming a strict separation between the lexicon and the syntax. The LFG architecture and the principle of Lexical Integrity dictate that fully inflected words are 'inserted' one-by-one into the c-structure, which does not seem to permit the blurring of the boundary between words and larger syntactic units that CG advocates. This paper addresses the question of how the intuitions behind constructions (in the CG sense) can be formalized within LFG, without rejection of the foundational assumptions behind the LFG framework. The key insight in our approach is the use of LFG templates (Dalrymple, Kaplan and King 2004, Crouch et al. 2007) to factor out grammatical information in such a way that it can be invoked by lexical items or by specific c-structure rules. C-structure rules that invoke specific templates are thus the equivalent of constructions in our approach, but Lexical Integrity and the separation of lexicon and syntax are preserved. However, there is a potentially deep theoretical consequence for the theory of the lexicon, because verbs on our approach specify default subcategorization through template calls in such a way that the subcategorization can be constructionally overridden. Thus, subcategorization is moved to the template component, which on our system is the interface between the lexicon and syntax.

It has been argued that the English *way*-construction (1) deserves a constructional analysis, rather than a compositional one, since the construction implies directed motion even though none of the individual words in *way*-examples necessarily denotes motion (Jackendoff 1990, Goldberg 1995).

(1) Sarah elbowed her way through the crowd.

A traditional LFG analysis could tie the sense of motion to one of the individual words (e.g., the verb or the word *way*) and appeal to a lexical redundancy rule to capture further restrictions on the construction. For example, we must account for the fact that *her way* is the f-structure OBJ but not the a-structure <theme> of *elbow*.

Toivonen (2002) discusses the English *way*-construction in connection with the Swedish Directed Motion Construction (2).

(2) Sarah armbågade sig genom mängden.
S. elbowed SELF through crowd.DEF

The Swedish DMC is very similar in meaning and use to the English *way*-construction, but the DMC does not include any word such as *way* to flag the construction. Instead, the construction is distinguished by the strict requirement of all constituents to be present, by certain restrictions on the individual constituents, and perhaps most interestingly, by a word order quirk at odds with the rest of Swedish grammar (Toivonen 2002). Consider (3a–b):

(3) a. Jonas knuffade sig in i mängden.
J. pushed SELF in inside crowd.DEF
'Jonas pushed his way into the crowd.'

b. Jonas knuffade in dig i mängden.
J. pushed in you inside crowd.DEF
'Jonas pushed you into the crowd.'

Verbal particles (such as *in*) are normally required to precede the direct object in Swedish, as in (3b). However, in the DMC, the particle follows *sig*, as in (3a). Toivonen's LFG analysis makes use of a lexical redundancy rule which alters the verb's argument structure and semantics, relating a 'regular' verb, e.g. *knuffa*, with a DMC version of that verb. All the pieces of the construction are thereby taken from the lexicon fully inflected and one-by-one, in accordance with Lexical Integrity. Toivonen's lexical rule effectively contains a construction, as the rule makes the requirement that each DMC verb include specified information about each of its syntactic arguments.

We propose here a single theory of constructions that uses existing LFG mechanisms such that a) commonalities between the English *way*-construction and the Swedish DMC are captured while b) preserving both the intuition that the *way*-construction is driven by lexical specifications for *way* together with general phrase structural facts about English and the intuition that the DMC is driven by a specific phrase-structural configuration.

Turning first to the DMC, we propose that this construction is most elegantly analyzed with the following construction-specific phrase structure rule, which makes crucial use of a template call:

(4) $V' \rightarrow$ V^0 (↑ = ↓, @DMC(↓ PRED FN)) NP ((↑ OBJ) = ↓, (↓ PRONTYPE) = SIMPLEX-REFLEXIVE) PP ((↑ OBL) = ↓)

We observe two important properties of (4). First, none of the phrase structure material in (4) is optional, since only this particular configuration has the special constructional meaning. In other words, our theory assumes that all optionality must be explicitly marked in phrase structure rules, as in computational LFG treatments (e.g. Crouch et al. 2007) and in contrast to theoretical positions that allow generalized optionality (e.g. Bresnan 2001). Second, we must explicitly state the fact that the NP must be a simplex reflexive, such as *sig*, and not just any kind of NP or even a complex reflexive (e.g. *sig själv*). An immediate benefit of this rule is that it correctly predicts that

the DMC reflexive cannot follow a particle (see (3a) above), because the particles in question are also assigned the grammatical function OBL (Toivonen 2003), which leads to a Uniqueness violation, given the obligatoriness of the OBL PP in (4) on the constructional reading.

The constructional template DMC is specified as follows:

(5) DMC(FN) = $(\uparrow \text{PRED}) = \text{`FN}\langle(\uparrow \text{SUBJ}),(\uparrow \text{OBJ}),(\uparrow \text{OBL})\rangle\text{'}$
$\lambda R\lambda x\lambda y\lambda P\lambda e\exists z\exists e'.\ R(e)\ \wedge\ agent(e) = x\ \wedge$
$traversal(e')\ \wedge\ agent(e') = x\ \wedge\ theme(e') = y\ \wedge$
$path(e') = z\ \wedge\ P(z)\ \wedge\ means(e') = e$:
$(\uparrow_\sigma \text{REL}) \multimap (\uparrow \text{SUBJ})_\sigma \multimap (\uparrow \text{OBJ})_\sigma \multimap (\uparrow \text{OBL})_\sigma \multimap (\uparrow_\sigma \text{EVENT}) \multimap \uparrow_\sigma$

The constructional template overrides the verb's default PRED feature and augments its default semantics (here represented as a Glue meaning constructor with event semantics as the meaning language).

This in turn means that the lexical entry for a verb cannot intrinsically specify the PRED and semantics. We assume that the verb *elbowed/armbågade*, which appears in (1) and (2), is specified as follows:

(6) elbowed/armbågade V $(\uparrow \text{PRED FN}) = \text{elbow}$
$\lambda e.elbow(e) : (\uparrow_\sigma \text{REL})$
$\left(\begin{array}{l} \text{@TRANSITIVE}(\uparrow \text{PRED FN}) \\ \lambda R\lambda x\lambda y\lambda e.R(e)\ \wedge\ agent(e) = x\ \wedge\ theme(e) = y: \\ (\uparrow_\sigma \text{REL}) \multimap (\uparrow \text{SUBJ})_\sigma \multimap (\uparrow \text{OBJ})_\sigma \multimap (\uparrow_\sigma \text{EVENT}) \multimap \uparrow_\sigma \end{array}\right)$

The verb specifies just its predicate FUNCTION (notice that this is not itself a semantic form, but rather part of a semantic form) and its s-structure semantic RELATION. The verb also optionally specifies its intrinsic semantics and calls the TRANSITIVE template. The TRANSITIVE template adds the verb's GF information as follows:

(7) TRANSITIVE(FN) = $(\uparrow \text{PRED}) = \text{`FN}\langle(\uparrow \text{SUBJ}),(\uparrow \text{OBJ})\rangle\text{'}$

The call to TRANSITIVE effectively serves as a default, because unless some other part of the system specifies an alternative, constructional GF template, there is no way to check Completeness and Coherence and the structure will fail.

The English *way*-construction is analyzed similarly, except that no exceptional phrase structure rule is required in this case. Rather, we assume the standard V′ rule for English, which already permits an NP OBJECT and a PP OBLIQUE. The locus of the construction is the word *way*, which receives the following specification:

(8) way N $(\uparrow \text{PRED}) = \text{`way'}$
$\{\ \lambda x.way(x) : (\uparrow_\sigma \text{VAR}) \multimap (\uparrow_\sigma \text{RESTR})\ \mid\ \text{@WAY}((\text{OBJ}\ \uparrow)\ \text{PRED FN})$ $\}$
$\lambda e\lambda x.traversal(e) \wedge theme(e) = x$:
$(\uparrow_\sigma \text{EVENT}) \multimap (\uparrow_\sigma \text{VAR}) \multimap (\uparrow_\sigma \text{RESTR})$

The left disjunct represents the non-constructional use of the word. The right disjunct represents the constructional use, which calls the WAY constructional template. It passes the template the PRED FN of the verb to which the *way*-NP serves as an object.

The constructional template for the *way*-construction is:

(9) WAY(FN) = $(\uparrow \text{PRED}) = \text{`FN}\langle(\uparrow \text{SUBJ}),(\uparrow \text{OBJ}),(\uparrow \text{OBL})\rangle\text{'}$
$\{\ \lambda R\lambda x\lambda Q\lambda P\lambda e\exists z\exists e'.\ R(e)\ \wedge\ agent(e) = x\ \wedge\ Q(e')\ \wedge\ agent(e') = x\ \wedge$
$path(e') = z\ \wedge\ P(z)\ \wedge\ means(e') = e$:
$(\uparrow_\sigma \text{REL}) \multimap (\uparrow \text{SUBJ})_\sigma \multimap ((\uparrow \text{OBJ})_\sigma \text{EVENT}) \multimap (\uparrow \text{OBJ})_\sigma \multimap$
$(\uparrow \text{OBL})_\sigma \multimap (\uparrow_\sigma \text{EVENT}) \multimap \uparrow_\sigma$ |

$\lambda R\lambda x\lambda Q\lambda P\lambda e\exists z\exists e'.\ R(e)\ \wedge\ agent(e) = x\ \wedge\ Q(e')\ \wedge\ agent(e') = x\ \wedge$
$path(e') = z\ \wedge\ P(z)\ \wedge\ manner(e') = R$:
$(\uparrow_\sigma \text{REL}) \multimap (\uparrow \text{SUBJ})_\sigma \multimap ((\uparrow \text{OBJ})_\sigma \text{EVENT}) \multimap (\uparrow \text{OBJ})_\sigma \multimap$
$(\uparrow \text{OBL})_\sigma \multimap (\uparrow_\sigma \text{EVENT}) \multimap \uparrow_\sigma$ $\}$

Notice that this template allows either a means interpretation for the construction or a manner interpretation. The latter interpretation is not possible for the DMC, although there is some variation that we will discuss.

In sum, we have captured the intuitions of Construction Grammar in LFG without giving up Lexical Integrity and without in any sense admitting constructions as first-class entities in the theory (unlike, e.g., the HPSG approach of Sag 1997 and certain subsequent HPSG work). LFG templates, which have been independently motivated for reasons of expediency in grammar writing, now play a crucial theoretical role: templates serve as the locus of grammatical information that can be either lexically or structurally invoked and thus formalize one aspect of the lexicon–syntax interface. In order to accommodate this view of constructions, the verbal lexicon needs to be modified such that subcategorization is now strictly governed by the template component.