C O P *i* L CAMBRIDGE OCCASIONAL PAPERS *in* LINGUISTICS
Volume **8**, Article 2: 23–40, 2015 | ISSN 2050-5949

THE SYNTAX-SEMANTICS INTERFACE IN GENERATIVE, PARALLEL AND FUNCTIONAL FRAMEWORKS

Lucía Contreras-García
University of Amsterdam

Abstract

This paper discusses the use of empty categories and/or mismatches in the representation of the syntactic and semantic structures of linguistic phenomena that infringe upon syntax-semantics interface transparency. Frameworks of the generative, parallel and functional type are discussed in relation to raising in order to establish the relationship between the architectural tenets that organize a given theory of language (the presence or absence of derivation between levels, the direction of derivations), on the one hand, and the licensing conditions for the use of empty syntactic categories and/or representational mismatches at the syntax-semantics interface, on the other.

1 Introduction

In this paper, I propose a comparison between the architecture, and in particular the syntax-semantics interface, of three notably different grammatical frameworks: Traditional Generative Grammar (henceforth TGG; Chomsky 1957, Chomsky 1957, Chomsky 1981, Chomsky 1993, Chomsky 1995, a.o.); the Parallel Architecture (henceforth PA; Jackendoff 1997, Jackendoff 2002, Jackendoff forthcoming, Culicover & Jackendoff 2005, a.o.); and Functional Discourse Grammar (henceforth FDG; Hengeveld 2004, Hengeveld & Mackenzie 2008, Hengeveld & Mackenzie 2010, a.o.) (for a more in-depth description and comparison, see Contreras-García 2013).

I will show the relationship between a given architecture of grammar, on the one hand, and the framework's tolerance of mismatches and/or empty categories at the syntax-semantics interface, on the other. It will be shown that there exist two distinct "tools" when accounting for linguistic phenomena that infringe upon syntax-semantics transparency, and that these are directly related to the architecture of a framework: representational mismatches and empty syntactic categories. The derivational and directional relation between the syntactic and the semantic levels of TGG, PA and FDG will be analyzed in order to illustrate the impact of their architecture upon their representation of violations of syntax-semantics transparency. In particular, it will be shown how these three frameworks account for raising, i.e. whether they use empty categories and/or mismatches at the syntax-semantics interface in such infringements of transparency and how this relates to their organization of grammar.

In section 2, I will introduce three features that define the architecture of a grammar: Distribution; Derivationality; and Directionality. In section 3, I will discuss syntax-semantics interface transparency vs. flexibility in relation to section 2 and the introduction of empty syntactic categories and/or syntax-semantics mismatches. In section 4, I will introduce the main architectural tenets of Traditional Generative Grammar, of the Parallel Architecture and of Functional Discourse Grammar and will relate them to sections 2 (their architectural tenets) and 3 (syntax-semantics interface transparency). In section 5, I will discuss how the three frameworks introduced in section 4 deal with a phenomenon that infringes upon syntax-semantics transparency, namely raising, and will relate this to sections 2 and 3. In section 6, I offer some conclusions.

2 Grammar in 3D

The architectural features that I will use to describe the design of grammatical frameworks are "the 3 Ds": Distribution; Derivationality; and Directionality (see Contreras-García 2013). Firstly, Distribution will be seen as the basic defining feature of the "map" of a grammar (the blueprint of a grammatical formalism). It determines what a model of grammar looks like and can be divided into: symbolic primitives; formation rule systems (see Pinker 1991, Lasnik 2000); levels of representation; and (uni and bi-) directional interfaces or mapping processes.

Secondly, Derivationality refers to the sequential, step-wise computational order in which a grammar accounts for linguistic structure, i.e. the calculation by means of which one level of representation can be translated into another one (Sadock 2003) such that the sequential steps of the derivation can be followed (Hale 1999). Derivation can take place between different grammatical (sub-) levels and it leads to derived levels being dependent upon the deriving level of representation. In a derivational framework, the source level of computation is formally responsible for all other levels. On the contrary, in an autonomous or parallel model of grammar all levels are simultaneous sources of linguistic computation such that no level precedes any other. In other words, in a derivational model of grammar well-formedness is determined by the correct formation of a sequence of steps in which things are added, deleted or moved around whereas in a non-derivational (constraint-based) model of grammar well-formedness is determined by the satisfaction of a set of non sequentially-ordered conditions (Jackendoff 1997: 12).

Thirdly, Directionality refers to the descriptive priority given to certain grammatical levels (Zwicky 1972), i.e. the direction that linguistic rules take in order to map structures through a grammatical system (Eliasson 1978: 50). If there is a hierarchical relation of some kind among the levels, the framework may be classified as either top-down (if functional levels precede formal levels) or bottom-up (if formal levels precede functional levels). Also, this architectural feature includes the uni or bi-directionality of inter-level mappings such that a directional model will show uni-directional interfaces (determining the direction of derivations) while a model of the parallel or autonomous type will show bi-directional interfaces (showing the non pre-determined direction of derivations).

3 The syntax-semantics interface in 3D

An inter-level interface is a formal interaction between two levels of representation. In a theory of language, an interface can be represented by means of a rule that determines correspondences between the various levels of representation ("linking algorithm" in Van Valin Jr. 2005: § 5.1). The syntax-semantics interface, i.e. the mapping between syntax and semantics, can be classified as either flexible or transparent. The 3 Ds determine the nature of a framework's syntax-semantics interface.

On the one hand, a transparent syntax-semantics interface is one that establishes a one-to-one relation between both levels of representation. In derivational models, in which there is a derivational relation between syntax and semantics, whatever happens in a derived level needs to be first of all representationally accounted for in the deriving level (if e.g. semantics is born from syntax, whatever happens in semantics needs to be accounted for first of all in syntax). A transparent syntax-semantics interface allows for a non-mismatching, straightforward correlation between both levels such that interface rules are in principle less complex than in non-derivational models. The "[s]emantic scope of constituents often depends on their syntactic constellation" and "the syntax-semantics interface (SSI) is iconic: Configurational asymmetries of syntactic tree structures are mapped onto semantic asymmetries" (Egg 2004). This means that the representation of a linguistic item in semantics can be easily tracked down to its correspondent representation at the syntactic level of representation. In order to provide all semantic elements with a syntactic counterpart, thus allow for a one-to-one mapping between syntax and semantics, "gaps" or "phonologically null pro-elements that have clear syntactic, semantic and/or morphological values" (Sadock 2012: 13) may be introduced - empty categories.

On the other hand, a flexible syntax-semantics interface is a mapping that may but need not per se allow for a straightforward correlation between syntax and semantics. This means that the representation of a linguistic item cannot always be easily tracked down to its correspondent representation(s) at a different level of representation, since there may be a deviation from the expected quantitative one-to-one correspondence, and/or from the expected qualitative default iconicity between the syntactic and the semantic level of representation. These deviations are called mismatches or "mappings between (apparently) incongruent elements or structures, where incongruity is defined relative to some typical or default condition" (Francis & Michaelis 2000). As opposed to the transparent interface of a derivational framework, the flexible interface of a non-derivational framework allows for inter-level mismatches, since levels do not necessarily relate in an iconic manner. In non-derivational frameworks, therefore, syntax-semantics interface rules play a very important role, since they are the ones to determine how syntax and semantics interact and, most importantly, the extent to which the syntactic and the semantic structures may differ and still interact in such a way that they represent one and the same linguistic unit.

4 3 Grammars in 3D

4.1 Traditional Generative Grammar

Traditional Generative Grammar (TGG) is considered here as it appears in Chomsky's Syntactic Structures (Chomsky 1957), the Standard Theory (Chomsky 1965), the Extended Standard Theory (Chomsky 1972), the Revised Extended Standard Theory (Chomsky 1975, Radford 1981), Government & Binding (Chomsky 1981, Radford 1988, Haegeman 1991) and the Minimalist Program (Chomsky 1993, 1995, Radford 1997, Hornstein, Nunes & Grohmann 2005). It holds for all these versions of TGG that their basic architecture consists of three levels of representation with their primitives (syntactic, semantic, phonological), which are defined by means of attribute-value pairs and hold structural relations between them. The phonological and the semantic components are purely interpretive (Chomsky 1965: 141) such that the generative capacity of language is attributed to syntax. Since the syntactic level is the source computational component, inter-level mapping processes are always born from syntax and target either the semantic or the phonological level. The basic design of TGG is illustrated in Figure 1 below.

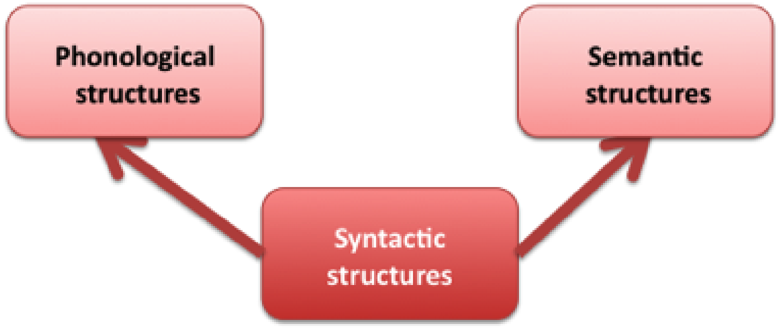

Figure 1 The basic architecture of Traditional Generative Grammar

Thus, the basic architecture of TGG leads to a "syntactocentric" framework of grammar (see e.g. Culicover & Jackendoff 2005), i.e. to a framework with an extremely powerful syntactic component from which phonology and meaning are derived (see Jackendoff 2010b). In later versions of the generative program (MP in Chomsky 1993, Chomsky 1995), D- and S-structures are eliminated, but Merge and Move constitute the main syntactic component from which meaning and phonology are still derived (see Burling 2003). Syntax remains the main computational component and the power of derivations is not only maintained but is even emphasized (Chomsky 1995: 362, see also Burling 2003, Zwart 1998). Note that this approach to inter-level relations is closely related to the overall Directionality of the model such that TGG results in a strongly directional model with uni-directional interfaces and only two possible mapping processes - from syntax into phonology and from syntax into semantics. As stated in section 3, such architecture of grammar has far-reaching consequences as to the type of syntax-semantics interface exhibited by the TGG framework. Since syntax is the source of all linguistic computation, it has to representationally account for anything that happens in semantics. The syntax-semantics interface is therefore transparent (one-to-one) and will avoid inter-level representational mismatches by introducing e.g. empty syntactic categories to provide semantic units with a syntactic counterpart.

4.2 The Parallel Architecture

The Parallel Architecture (PA) is considered here as it appears in Jackendoff (1983, 1990, 1997, 1999, 2002, 2007, 2010a,b, forthcoming) and in Culicover & Jackendoff (2005). In PA, each level has its own set of units and principles of combination such that syntax, semantics and phonology are three independent generative systems with their own combinatorial power (Jackendoff 1999: 395), (Jackendoff forthcoming: 9) that are processed parallelly and linked by interfaces stipulating how the different types of structures may correlate (Jackendoff 2007: 5). Units of phonological and semantic structure cannot be directly derived from syntactic structures such that syntactic, semantic and phonological principles are exclusively specific to the syntactic, semantic and phonological levels respectively.

The phonological structure (PS) can include the sequence of phonemes and syllables, the stress of a phrase and the morphophonology of sequences (whether it is a phonological word, phrase, affix, an utterance, etc). The semantic level is called the conceptual structure (CS, Jackendoff 1983, Jackendoff 1990, Jackendoff 2002). The syntactic structure (SS) consists of trees that look like the traditional ones but that are not binary-branching and have no words at the bottom (in an attempt to keep all non strictly syntactic information out of SS). The constraint on the SS is such that all parts of the final tree have to conform to one of the treelets (pieces of stored structure that only consists of syntactic variables). Treelets are then clipped together at shared nodes ("unification" as in Shieber 1986, Jackendoff forthcoming: 8). There is no specific order to build syntactic trees such that syntactic tree building is compatible with serial, parallel, top-down and bottom-up computation (Jackendoff 1997: 8-9, Jackendoff 2007: 11).

Also, since phonology and semantics are not derived from syntax, "ordered derivations are replaced by parallel constraint checking" (Jackendoff forthcoming: 11), plus the correct activation of 'correspondence rules' or the 'interface component' " that regulates the ways in which the syntactic, semantic and phonological structures may interact (Jackendoff 1999: 395). "The structures of each component are licensed by simultaneously applied component-internal constraints" and "[t]he relationships among structures in different components are licensed by interface constraints" (Jackendoff 2010b: 588). This is called "representational or structure-based modularity", which creates a system in which "[e]ach separate form of representation has its own particular autonomous (i.e., domain-specific) structure, and its own interfaces to other structures" and in which "[o]ne form of representation is relatively informationally encapsulated from another to the degree that one can influence the other only through a series of interfaces, or through a narrowly specialized interface" (Jackendoff 2010b: 586). The direction of interfaces is therefore not pre-determined (it is inherently nondirectional, see Jackendoff 2007: 5) such that no specific level needs to be the main computational source. Interfaces start from, and target, all levels of representation. This is why arrows between levels are bi-directional (to express possible correlations between levels rather than derivation among them). PA is thus all-directional, although constraints in PA can be implemented in a specific order so as to fit specific processing tasks (Jackendoff 2010a: 5). An architectural overview of PA is offered in Figure 2 below.

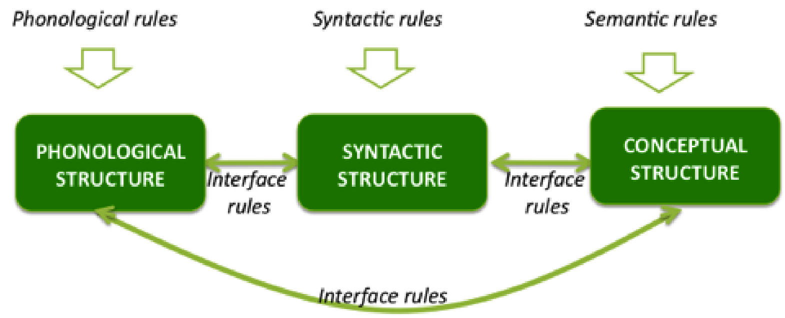

Figure 2 The basic architecture of The Parallel Architecture

Therefore, the Parallel Architecture goes against the notion of logical sequence typical of a syntactocentric derivation (Jackendoff 2010a: 4-5). Whereas TGG represents the syntactocentric assumption that "everything begins with syntax", PA represents "a radical dethronement of syntax from its ruling position" (Burling 2003). There is no derivation between the levels of representation but rather all levels may be both the source and the target of an interface process. PA is a fully autonomous or non-derivational framework, which has consequences upon its syntax-semantics interface. Since semantics and phonology have their own generative and combinatorial capacity, syntax is simpler - the minimum syntax required to mediate between phonology and semantics (Culicover & Jackendoff 2005). The syntax-semantics interface is flexible and can thus show mismatches (it does not have to be maximally simple and uniform). Linguistic phenomena that infringe upon syntax-semantics transparency e.g. that present meaning in the absence of form do not need to resort to empty syntactic material in order to provide all semantic material with a syntactic counterpart and thus observe a one-to-one correspondance between syntax and semantics.

4.3 Functional Discourse Grammar

Functional Discourse Grammar (FDG) is considered here as it appears in Hengeveld (2004), Mackenzie & Gómez-González (2005), Hengeveld & Mackenzie (2008, 2010). FDG is a further development of Functional Grammar (Dik 1978, 1997a,b). It is a structural-functional theory of language (Butler 2003), i.e. it positions itself between strictly formal and strictly functional theories. In FDG, linguistic phenomena are represented by means of "multiple orthogonal representations of linguistic phenomena" (Hengeveld & Mackenzie 2008: 31) that are related by means of formulation and encoding rules (Bakker 2001, 2005). It is composed of four hierarchically-organized grammatical levels: the pragmatic or interpersonal level (IL); the semantic or representational level (RL); the morphosyntactic level (ML); and the phonological level (PL). The interpersonal and representational levels are coded by the operation of Formulation while the morphosyntactic and phonological levels are coded by the operation of Encoding. In FDG, each level of representation possesses unique, distinctive units, i.e. units that are not derivable from other levels (see Hengeveld 2004: 5-8). In Formulation, formation rules make use of primitives containing frames, lexemes and operators. In morphosyntactic Encoding, formation rules make use of templates, grammatical morphemes and morphosyntactic operators. In phonological Encoding, formation rules make use of templates, suppletive forms and phonological operators. The various primitives are then combined by formation rules in order to produce the various levels of representation (Hengeveld & Mackenzie 2008: 12, 13, 19). The basic architecture of Functional Discourse Grammar is presented in Figure 3 below.

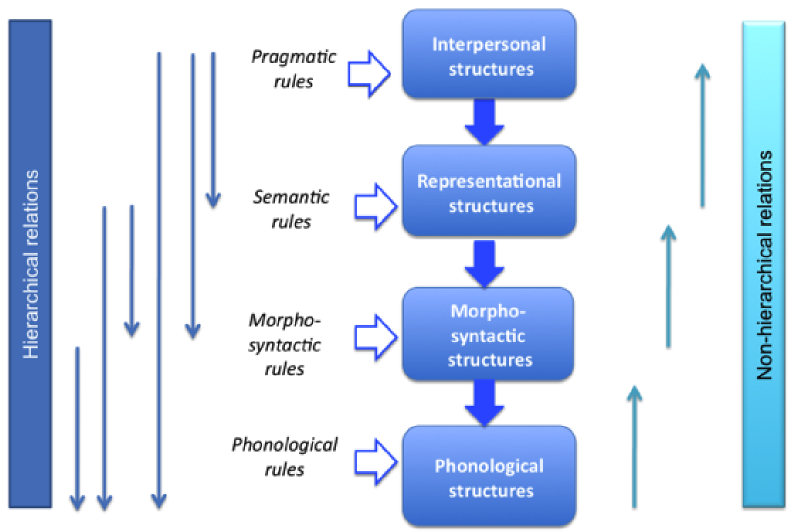

Figure 3 The basic architecture of Functional Discourse Grammar

The FDG architecture shows a hybrid, non-default combination of features regarding the 3 Ds discussed in section 2. On the one hand, each level of representation contains units and information that is specific to that one level. Also, rules are specific to each level of representation, i.e. levels are not derived from each other. Pragmatics and semantics may be computed in tandem (Hengeveld & Smit 2009) and lower levels may start computing as soon as enough information is fed by higher levels (incremental processing or "depth first principle", Hengeveld & Mackenzie 2008: 23-25). These characteristics are, default case-scenario, typical of a non-derivational grammatical framework. However, FDG shows a strong top-down directionality, since representations from higher function levels (pragmatics and semantics) are translated into representations at lower formal levels (morphosyntax and phonology) (Hengeveld 2004: 3, Hengeveld & Mackenzie 2008: 1-3). There is a main computational component, since computation always starts with pragmatics (with the exception of non-hierarchical, bottom-up relations, see Hengeveld & Smit (2009) such that the model is strongly "pragmato-semantocentric" (Hengeveld p.c.) - or just pragmatocentric, since the representation of certain linguistic elements can lack a semantic representation. These characteristics are, default case-scenario, typical of a derivational framework of grammar.

FDG's hybrid approach to the syntax-semantics interface is, once more, a direct consequence of its hybrid architecture, i.e. its hybrid approach to the 3 Ds. Specifications at the syntactic and the semantic level do not obligatorily need to show a one-to-one mapping - representational mismatches arise. Although regular correspondences (a transparent interface) between the different levels are preferred, syntax and semantics are independent from each other and a wide variety of interfaces is possible (Hengeveld & Smit 2009: 16) (a flexible interface). Inter-level independence is reflected in that a single semantic constituent can acquire different representations at the syntactic level (and vice versa), and in that only those aspects that are relevant to build up the semantic or the syntactic level are used for the representation (Hengeveld & Smit 2009: 22, 25). In section 5, I will show that the hybrid architecture of FDG translates into a mixed use of theory-driven devices such as empty syntactic categories in the representation of linguistic phenomena that infringe upon syntax-semantics transparency, aligning itself with derivational theories of language (such as TGG), and of syntax-semantics mismatches, aligning itself with with parallel theories of grammar (such as PA).

5 Raise your hands and don't move

5.1 Raising the syntax-semantics interface into a 3D level

Raising involves an infringement upon syntax-semantics transparency. In raising, a semantic argument of the embedded clause corresponds to a syntactic element of the matrix clause that has supposedly been raised from the subject or object position of the embedded clause into the matrix clause. (1) below illustrates an instance of raising. From a syntactic point of view, (1a) shows that raising is the subject of the matrix clause (MC). From a semantic point of view, (1b) shows that raising is the external argument of the embedded clause (EC). This creates a syntax-semantics mismatch.

(1) a. [$_{\text{MC}}$ *raising*$_{\text{Subj}}$ seems [$_{\text{EC}}$ ∅ to violate syntax-semantics transparency]]
b. [$_{\text{MC}}$ ∅ seems [$_{\text{EC}}$ *raising*$_{\text{Agent}}$ violates syntax-semantics transparency]]

Since raising results in a syntax-semantics mismatch, a theory of language has two options: accounting for such a mismatch by means of a discrepancy between the syntactic and the semantic representations (thus creating a flexible syntax-semantics interface); or avoiding the discrepancy in order to provide the semantic argument with a syntactic counterpart in the embedded clause (thus creating a transparent syntax-semantics interface). As discussed in the previous sections, the option is determined by the architectural pillars of the theory at hand (see also Contreras-García 2013 for a more in-depth description of § 5).

5.2 Raising in Traditional Generative Grammar

In the generative framework, (subject) raising may be accounted for by means of a movement procedure whereby the raised element is born at the subject position of the embedded clause and is then raised into the subject position of the matrix clause, leaving a co-indexed trace behind. This is illustrated in Figure 4 for (2) below. The dotted rectangle draws attention to a phonetically null category representing the subject of the embedded clause before being moved up.

(2) This seemed to be explanation enough.
(Butler 1964: 88, first published 1964)

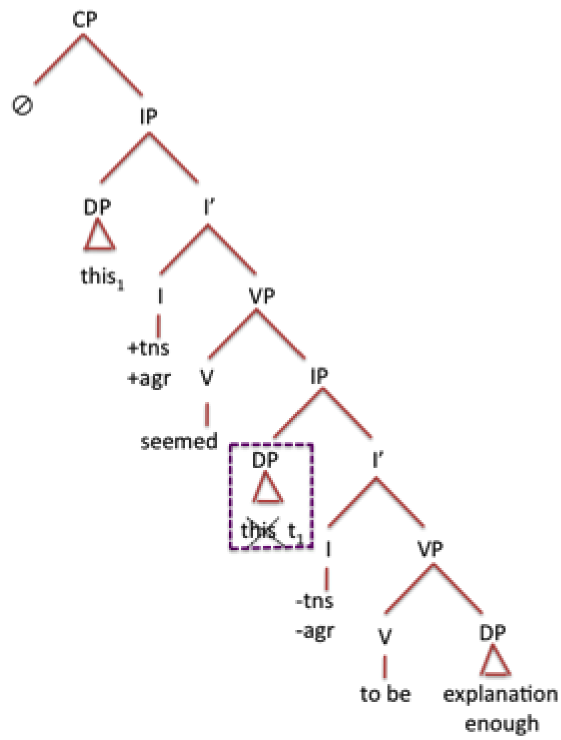

Figure 4 Raising in Traditional Generative Grammar

As shown in Figure 4 above, "this" is born in the subject position of the embedded clause (Spec IP) and subsequently moves up to the subject position of the matrix clause. The co-indexed trace t_1 (also called the tail, what is left behind when the subject is raised) in the embedded clause is bound to its antecedent head $this_1$ in the matrix clause (see e.g. Hornstein 1999). The subject can alternatively be located at Spec VP (and subsequently be raised to Spec IP via subject raising) according to the VP-internal subject hypothesis (see Koopman & Sportiche 1991, Radford 1997, Burton & Grimshaw 1992). Theta-role distribution takes place before movement such that it is the external argument of the embedded clause, and not that of the matrix clause, that receives the role of undergoing "to be explanation enough". Note also that an MP account (e.g. Chomsky 1995, see also Polinsky & Potsdam 2006) would maintain that "this" is not raised from the embedded up into the matrix clause but rather copied from the embedded into the matrix clause such that only the highest copy would be spelled out.

To sum up, TGG accounts for the mismatch created by subject raising such that a syntax-semantics discrepancy is avoided in the representation. The raised element is born within the embedded clause and is then moved up into the matrix clause, leaving a trace behind. Alternatively, it is copied from the lower into the higher position such that only the higher copy is read at spell out. In both analyses, there is a one-to-one mapping between syntax and semantics in that the external argument of the embedded clause is provided with a syntactic counterpart. The interface created is a transparent one. This is in keeping with a generative framework in which semantics is derived from syntax such that what happens at the semantic level (the external argument of the embedded clause) needs to be accounted for first of all at the syntactic level (an empty category, be it a trace or a copy).

5.3 Raising in the Parallel Architecture

In the Parallel Architecture, raising is represented by means of a mismatch between the syntactic and the semantic structures. This is illustrated in Figure 5 for (3) below.

(3) This seemed to be explanation enough. (=(2))

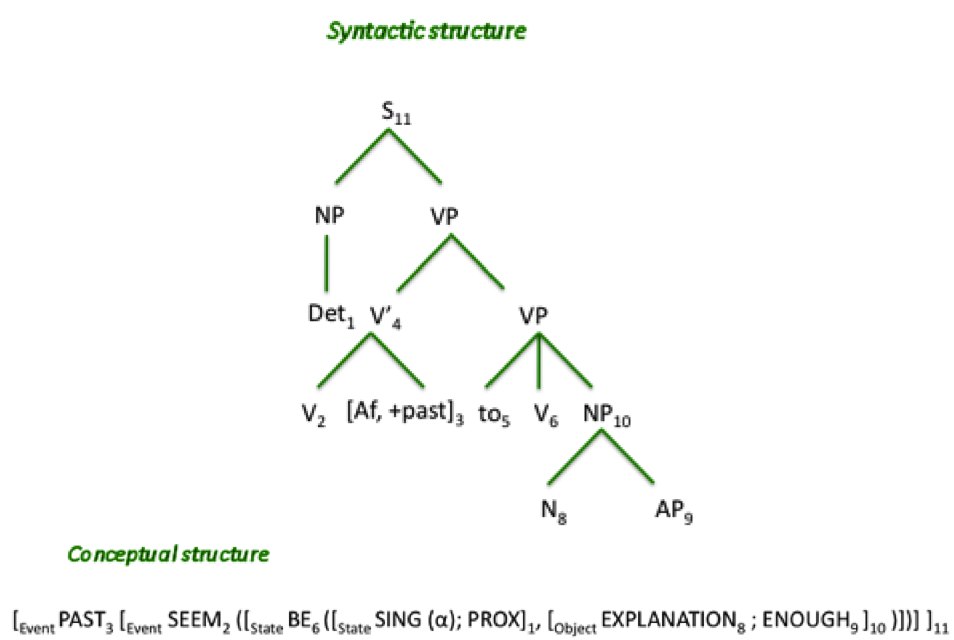

Figure 5 Raising in the Parallel Architecture

Figure 5 above shows that in the syntactic structure (SS) of the [NP[Det_1]] “this” appears as a sister of the higher VP and not as that of the embedded clause (i.e. “this” is higher up in the syntactic tree than “seem”). The conceptual structure (CS) is such that SEEM has scope over $[_{\text{State}}\text{SING } (\alpha);\ \text{PROX}]_1$ (which represents “this” at CS and is therefore co-indexed with Det_1 at SS). Therefore, PA accounts for the mismatch created by subject raising such that the syntax-semantics discrepancy is not avoided in the representation. The external argument of the embedded clause is represented within the matrix clause in the syntactic structure but within the embedded structure in the conceptual structure. In this analysis, there is not a one-to-one mapping between syntax and semantics because the external argument of the embedded clause is not provided with a syntactic counterpart[1]. The interface created between the syntactic and the semantic structures is therefore a flexible one. This is in keeping with a parallel framework in which semantics is not derived from syntax such that what happens at the semantic level (the external argument of the embedded clause) needs not be accounted for at the syntactic level by means of an empty syntactic category.

5.4 Raising in Functional Discourse Grammar

In Functional Discourse Grammar, subject raising is represented by means of a mismatch between the pragmatic, the semantic and the morphosyntactic structures (see Hengeveld & Mackenzie 2008: 367-372). This is illustrated in Figure 6 for (4) below. The dotted rectangle indicates the empty category that is introduced to provide the external argument of the embedded clause with a morphosyntactic counterpart.

(4) This seemed to be explanation enough. (=(3))

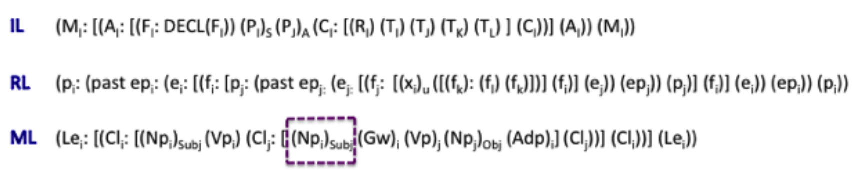

Figure 6 Raising in Functional Discourse Grammar

1 Note that PA also has a grammatical function (GF) tier that represents ranked NP arguments with a grammatical function in a given utterance. The grammatical function tier in PA for (3) would be $[\text{GF}_1]_2[\text{GF}_1 > \text{GF}_{10}]_6$ such that GF_1 (for “this”) would appear twice, once under the scope of “seem” (this GF would be SS-linked) and once under the scope of “be” (this GF would be CS-linked).

As illustrated in Figure 6 above, the Interpersonal Level (IL) contains one referential subact R_1 for "this". At the Representational Level (RL), "this" is represented by $(x_i)_U$, under the scope of "seem". At the Morphosyntactic Level, "this" receives two co-indexed $(NP_i)_{Subj}$, one within the matrix clause (Cl_i) and one within the embedded clause (Cl_j) (see Hengeveld & Mackenzie 2008: 372). Since the first $(NP_i)_{Subj}$, is the only overt subject, the second $(NP_i)_{Subj}$ is empty ($\emptyset$).

FDG accounts for the mismatch created by raising such that the discrepancy is not avoided in the representation. The external argument of the embedded clause is represented both within the matrix clause and within the embedded clause in the syntactic structure but only once, within the embedded structure, in the conceptual structure, and only once in the pragmatic or interpersonal structure. In this analysis, there is no one-to-one mapping between meaning and form, yet the external argument of the embedded clause is provided with an empty syntactic counterpart. The introduction of an empty syntactic category that goes unpronounced (typical of a derivational framework), plus the mismatch created between the representation of meaning and form structures (typical of a non-derivational framework) is in keeping with a model of grammar with a hybrid approach to the 3 Ds.

6 Conclusions

In this paper, I have discussed the architecture of Traditional Generative Grammar (cf. Chomsky 1957, 1965, 1981, 1993, 1995), the Parallel Architecture (cf. Jackendoff 1997, 2002, forthcoming, Culicover & Jackendoff 2005) and Functional Discourse Grammar (cf. Hengeveld 2004, Hengeveld & Mackenzie 2008, 2010) according to various metatheoretical parameters, in particular the relationship between their levels of representation (derivational vs. non-derivational, directional vs. parallel framework). I have then related this to the various frameworks' syntax-semantics interface. In particular, I have discussed how a framework's architectural features determine whether its syntax-semantics interface is transparent (thus avoids mismatches between its representation of the syntactic and semantic level) or whether its syntax-semantics interface is flexible (thus does not avoid such mismatches). The transparent or flexible nature of a given framework's syntax-semantics interface is best seen in phenomena that infringe upon syntax-semantics transparency such as ellipsis or raising. I have illustrated how Traditional Generative Grammar, the Parallel Architecture and Functional Discourse Grammar deal with raising. I have observed that there is a close relation between a framework's architecture and the way in which its syntax-semantics interface approaches raising's non-observance of syntax-semantics transparency. Firstly, Traditional Generative Grammar, a typical derivational and directional framework whereby semantics is derived from syntax, accounts for raising by resorting to an empty syntactic category that provides all semantic material with a syntactic counterpart such that the syntax-semantics interface is one-to-one (it is transparent). Secondly, the Parallel Architecture, a typical non-derivational and non-directional (parallel) framework whereby semantics is not derived from syntax, accounts for raising by means of a representational mismatch between the syntactic and sematic structures such that not all semantic material is provided with a syntactic counterpart and the syntax-semantics interface is not necessarily one-to-one (it is not necessarily transparent). Thirdly, Functional Discourse Grammar, a hybrid framework that shares architectural features typical of derivational and directional models and features of typical non-derivational and non-directional models of grammar, accounts for raising by resorting to both an empty syntactic category and a mismatch between the representation of the semantic and syntactic structures. To sum up, this paper has demonstrated the existing relationship between a framework's architecture regarding derivation and direction, its syntax-semantics interface, its use of mismatches between the syntactic and semantic structures and its use of empty syntactic categories in the representation of linguistic phenomena that infringe upon syntax-semantics transparency.

References

Bakker, D. 2001. The FG expression rules: a dynamic model. *Revista Canaria de Estudios Ingleses* 42. 15–54.

Bakker, D. 2005. Expression and agreement: some arguments for the dynamic expression model. In H. Hengeveld & C. de Groot (eds.), *Morphosyntactic expression in functional grammar*, 1–40. Berlin/New York: Mouton de Gruyter.

Burling, R. 2003. Review of R. Jackendoff (2002), Foundations of Language. Brain, Meaning, Grammar, Evolution. Oxford/New York: Oxford University Press. *Language in Society* 32. 415–418.

Burton, S. & J. Grimshaw. 1992. Coordination and VP-internal subjects. *Linguistic Inquiry* 23(2). 305–313.

Butler, C. 1964. *Read and listen. los mejores relatos del S. XX en V.O. integra.* Madrid: PONS 2010th edn.

Butler, C. 2003. *Structure and function: A guide to three major structural-functional theories.* Amsterdam/Philadelphia: John Benjamins.

Chomsky, N. 1957. *Syntactic structures.* The Hague: Mouton.

Chomsky, N. 1965. *Aspects of the theory of syntax.* Cambridge: MIT Press.

Chomsky, N. 1972. *Studies on semantics in generative grammar*. The Hague: Mouton.
Chomsky, N. 1975. *Reflections on language*. New York: Pantheon.
Chomsky, N. 1981. *Lectures on government and binding*. Dordrecht: Foris.
Chomsky, N. 1993. A minimalist program for linguistic theory. In K. Hale & S. J. Keyser (eds.), *The view from Building 20. Essays in linguistics in honor of Sylvain Bromberger*, Cambridge: MIT Press.
Chomsky, N. 1995. *The minimalist program*. Cambridge: MIT Press.
Contreras-García, L. 2013. *Grammar in 3D: On linguistics theory design*: University of Amsterdam PhD dissertation.
Culicover, P.W. & R. Jackendoff. 2005. *Simpler syntax*. Oxford/New York: Oxford University Press.
Dik, S. C. 1978. *Functional grammar*. Amsterdam: North-Holland.
Dik, S. C. 1997a. *The theory of functional grammar. Part I. the structure of the clause*. Berlin/New York: Mouton de Gruyter.
Dik, S. C. 1997b. *The theory of functional grammar. Part II: Complex and derived constructions*. Berlin/New York: Mouton de Gruyter.
Egg, M. 2004. Mismatches at the syntax-semantics interface. In Müller (ed.), *Proceedings of the 11th international conference on Head-Driven Phrase Structure Grammar*, 119–139. Stanford: CSLI Publications.
Eliasson, S. 1978. Directionality in lexical signs and grammars. remarks on the emergence of a theoretical concept. *Studia Linguistica* 32. 50–62.
Francis, E. J. & L. A. Michaelis. 2000. Approaches to mismatch. Introduction. In M. Butt & T. Hollway King (eds.), *Proceedings of the BFG00 conference workshops*, Stanford: CSLI Publications.
Haegeman, L. 1991. *Introduction to government and binding theory*. Oxford/Cambridge: Blackwell Publishers 1994th edn.
Hale, J. 1999. Representational versus derivational approaches to syntax. optimality theory and chains. (unpublished manuscript) Available online at http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.37.1827.
Hengeveld, K. 2004. *A new architecture for functional grammar* chap. The architecture of a Functional Discourse Grammar, 1–21. Berlin/New York: Mouton de Gruyter.
Hengeveld, K. & J. L. Mackenzie. 2008. *Functional discourse grammar. A typologically-based theory of language structure*. Oxford/New York: Oxford University Press.
Hengeveld, K. & J. L. Mackenzie. 2010. Functional Discourse Grammar. In B. Heine & H. Narrog (eds.), *The Oxford handbook of linguistic analysis*, Oxford/New York: Oxford University Press.
Hengeveld, K. & N. Smit. 2009. Dynamic formulation in functional discourse

grammar. *Lingua* 119(8). 1118–1130.

Hornstein, H. 1999. Movement and control. *Linguistic Inquiry* 30(1). 69–96.

Hornstein, N., J. Nunes & K. Grohmann. 2005. *Understanding minimalism.* Cambridge: Cambridge University Press.

Jackendoff, R. 1983. *Semantics and cognition.* Cambridge: MIT Press.

Jackendoff, R. 1990. *Semantic structures.* Cambridge: MIT Press.

Jackendoff, R. 1997. *The architecture of the language faculty* (Linguistic Inquiry Monographs 28). Cambridge: MIT Press.

Jackendoff, R. 1999. Parallel constraint-based generative theories of language. *Trends in Cognitive Sciences* 3(10). 393–400.

Jackendoff, R. 2002. *Foundations of Language. Brain, Meaning, Grammar, Evolution.* Oxford/New York: Oxford University Press.

Jackendoff, R. 2007. A Parallel Architecture perspective on language processing. *Brain Research* 1146. 2–22.

Jackendoff, R. 2010a. *Meaning and the lexicon. The Parallel Architecture 1975-2010.* Oxford/New York: Oxford University Press.

Jackendoff, R. 2010b. The Parallel Architecture and its place in cognitive science. In B. Heine & H. Narrog (eds.), *Oxford handbook of linguistic analysis*, 583–605. Oxford/New York: Oxford University Press.

Jackendoff, R. forthcoming. Language. To appear in 'Cambridge Handbook to Cognitive Science'. Available online at http://ase.tufts.edu/cogstud/incbios/RayJackendoff/recentpapers.htm.

Koopman, H. & D. Sportiche. 1991. The position of subjects. *Lingua* 84. 211–258.

Lasnik, H. 2000. *Syntactic structures revisited.* Cambridge: MIT Press.

Mackenzie, J. L. & M.Á Gómez-González. 2005. *Studies in functional discourse grammar.* Bern/Berlin/Bruxelles/Frankfurt am Main/New York/Oxford/Wien: Peter Lang.

Pinker, S. 1991. Rules of language. *Science* 253(5019). 530–535.

Polinsky, M. & E. Potsdam. 2006. Expanding the scope of control and raising. *Syntax* 9. 171–192.

Radford, A. 1981. *Transformational syntax. A student's guide to Chomsky's Extended Standard Theory.* Cambridge: Cambridge University Press.

Radford, A. 1988. *Transformational grammar. A first course.* Cambridge: Cambridge University Press.

Radford, A. 1997. *Syntax. A Minimalist introduction.* Cambridge: Cambridge University Press.

Sadock, J. M. 2003. Mismatches in autonomous modular versus derivational grammars. In E. J. Francis & L. A. Michaelis (eds.), *Mismatch - form-function incongruity and the architecture of grammar*, 335–357. Chicago:

University of Chicago Press.

Sadock, J. M. 2012. *The modular architecture of grammar*. Cambridge: Cambridge University Press.

Shieber, S. 1986. *An introduction to unification-based approaches to grammar*. Stanford: CSLI Publications.

Van Valin Jr., R. D. 2005. *Exploring the syntax-semantics interface*. Cambridge: Cambridge University Press.

Zwart, C. J. W. 1998. Review of N. C.homsky (1995). The Minimalist Program. Cambridge: MIT Press. *Journal of Linguistics* 34. 213–226.

Zwicky, A. M. 1972. Remarks on directionality. *Journal of Linguistics* 8. 103–109.

Lucía Contreras-García
University of Amsterdam

luciaolucia@yahoo.es