Grammar in 3D: on linguistic theory design
Contreras García, L.

Citation for published version (APA):
Contreras García, L. (2013). Grammar in 3D: on linguistic theory design

General rights
It is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), other than for strictly personal, individual use, unless the work is under an open content license (like Creative Commons).

Disclaimer/Complaints regulations
If you believe that digital publication of certain material infringes any of your rights or (privacy) interests, please let the Library know, stating your reasons. In case of a legitimate complaint, the Library will make the material inaccessible and/or remove it from the website. Please Ask the Library: http://uba.uva.nl/en/contact, or a letter to: Library of the University of Amsterdam, Secretariat, Singel 425, 1012 WP Amsterdam, The Netherlands. You will be contacted as soon as possible.
6  FINAL REMARKS AND CONCLUSIONS

“Any physical theory is always provisional, in the sense that it is only a hypothesis: you can never prove it. No matter how many times the results of experiments agree with some theory, you can never be sure that the next time the results will not contradict the theory. On the other hand, you can disprove a theory by finding even a single observation that disagrees with the predictions of the theory. As philosopher of science Karl Popper has emphasized, a good theory is characterized by the fact that it makes a number of predictions that could in principle be disproved or falsified by observation. Each time new experiments are observed to agree with the predictions the theory survives, and our confidence in it is increased; but if ever a new observation is found to disagree, we have to abandon or modify the theory. At least that is what is supposed to happen, but you can always question the competence of the person who carried out the observation” (Hawking 1988: 10).

Metatheory

Doing metatheory is doing theory on other theories. In linguistics, doing metatheory is doing theory on other linguistic theories. A linguistic theory attempts to model language by means of symbols and establishing relations among those symbols. By doing metatheory on linguistic theories, therefore, I have attempted to theorize on different formalizations and ways in which various views of language intend to best “translate” the product and/or process of language into a formal system. By speaking about language and the translation of language into a formal system, a clear distinction is made between the object of study of linguistics -language- and the method to study that

Grammar in 3D

In order to establish the features that define the design of grammars, I have adopted a 3D approach by establishing three macro-features that include several relevant characteristics of the architecture of any grammar: Distribution, Derivation and Direction. These I have divided into further architectural sub-parameters with which I have analyzed and compared Traditional Generative Grammar, the Parallel Architecture and Functional Discourse Grammar. I have then related distributional, derivational and directional features to each other in general, to each of the models in particular and to the use of empty categories and similar devices.

Distribution

The first architectural dimension, Distribution, has been defined as the basic design feature of the architecture of a grammar. Distribution determines what a model of grammar looks like and has been further divided into symbolic primitives and formation rule systems (see Pinker 1991, Lasnik 2000), dependent and independent levels of representation, and uni and bi-directional
inter-level mapping processes. All these architectural features are like the pieces of a patchwork: put together in a particular way, they make up the map of a grammar.

In Traditional Generative Grammar, independent syntactic rules give birth to an independent level of syntax. There are two further, non-autonomous levels: semantics and phonology. The grammar is thus made out of a syntactic component, a semantic component, and a phonological component, the latter two being purely interpretive (Chomsky 1965: 141) such that the generative capacity of language is attributed to syntax. Since the syntactic level is the main computational one, inter-level mapping is always born from it and targets either the semantic or the phonological level.

In the Parallel Architecture, each form of representation is born from its formation rules and has its own, autonomous structure such that information is relatively encapsulated and information from one level influences that from other levels by interfaces. This is called “representational or structure-based modularity” (Jackendoff 2010b: 586, italics in original). Syntax, semantics and phonology are thus all independent generative systems (Jackendoff 1999: 395). Since all levels are independent, inter-level mapping processes are all potential source and target levels of computation.

In Functional Discourse Grammar, four levels of representation are computed from independent formation rules. In Formulation (for the pragmatic and the semantic level), 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) such that, although regular correspondences between the different levels are preferred, levels are independent from each
other and a wide variety of interfaces is possible (ibid: 16). Inter-level independence is reflected in that a single constituent can acquire different representations at the various levels making use of different sets of primitives, and in that only those aspects that are relevant to build up a certain level or an aspect of an utterance are used for the representation (ibid: 22, 25). This means that all inter-level mapping processes should in principle be allowed unless a non-default combination of distributional, derivational and directional features takes place.

**Derivation**

The second architectural dimension, Derivation, has been considered as the sequential, step-wise computational order in which a grammar accounts for linguistic structure, as the calculation by means of which one can translate one level into another level (Sadock 2003). Derivation can take place between different levels of representation or between different strata of a single level. Within Derivation, levels of representation have been classified into source levels, target levels, source-and-target levels, and all-first source levels (compulsorily initial levels of computation). 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. In a non-derivational (constraint-based) model of grammar, a set of non sequentially-ordered conditions need to be satisfied (Jackendoff 1997: 12). The answers to the following questions have shown a strong correlation with a model’s approach to Derivation: Are all levels source levels? Are all source levels also target levels? Are all source-and-target levels also possibly all-first levels of linguistic structure computation?

An architectural feature that has a direct impact upon the approach that a theory of language adopts regarding Derivation is the nature of inter-level mapping processes or interfaces. Interfaces have been classified as either flexible or transparent.
A transparent interface is an isomorphic interface between different levels of representation. A flexible interface, on the other hand, is an interface that allows for inter-level non-homomorphisms. Mismatches are “mappings between (apparently) incongruent elements or structures, where incongruity is defined relative to some typical or default condition” (Francis & Michaelis 2000). I have classified mismatches according to two main, cross-cutting parameters: inter-level vs. intra-level mismatches; and quantitative vs. qualitative mismatches (mismatches in the number of elements at different levels vs. mismatches in the (expected) type of elements at different levels).

Accordingly, traditional Generative Grammar avoids mismatches between say the syntactic and the semantic levels since it is a derivational model in which the semantic level is read from the syntactic one. This means that an element at the semantic level has to also representationally be accounted for at the syntactic level, since the former is born from the latter. This creates a transparent interface. The computational power of the syntactic level and its far-reaching architectural consequences has been called “syntactocentrism” – syntax as the starting computing mechanism from which phonology and meaning are derived (Jackendoff 2010b). In later versions of the generative program (MP in Chomsky 1993, 1995 a.o.), 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).

In the Parallel Architecture, constraints substitute derivations such that the well-formedness of a final structure is determined by the well-formedness of the various, independent structures and the correct application of interface rules. Thus, level-internal constraints regulate relations within levels whereas level-external constraints regulate relations between levels, i.e. the ways in which the semantic, syntactic and phonological structures may interact - the ‘correspondence rules’ or the ‘interface component’ (Jackendoff 1999: 395). A parallel
derivation goes against the notion of logical sequence typical of a syntactocentric derivation (Jackendoff 2010a: 4-5). There is thus no derivation between the levels of representation such that all levels may both be source and target of an interface process. In the presence of such a non-derivational model of grammar, mismatches are allowed in that no level is fully responsible for a different level in its representation. This creates a flexible approach to inter-level mapping processes.

In Functional Discourse Grammar, linguistic phenomena are represented by means of multiple and mismatching orthogonal representations at the various levels of representation (Hengeveld & Mackenzie 2008: 5, 26), which are related by means of formulation and encoding rules (Bakker 2001, 2005). The fact that the pragmatic, semantic, morphosyntactic and phonological levels are independent means that the model should be non-derivational – i.e. that no level is derived from another level and that all levels may be either source or target sources of formal computation. However, the model is strongly “pragmato-semantocentric” (Hengeveld p.c.) – or just pragmatocentric, since the representation of certain elements without semantic content may lack a semantic level at all in that computation always proceeds from function into form – with the exception of non-hierarchical, bottom-up relations (see Hengeveld & Smit 2009). That the model is pragmato(-semanto)centric, in turn, means that not all levels may be source and target of computation, thus the nature of the model is not all-directional but rather a specific sequence must take place in the implementation of the formalism. On the other hand, since all levels are relatively autonomous, mismatches may arise and the nature of interfaces is flexible.

I have further stated a default relation between a model’s choice of distributional features, on the one hand, and its choice of derivational features, on the other. If a model of grammar possesses independent formation rules and sets of primitives for each of its levels of representation, the tendency is that all levels will be autonomous and be potential sources and targets of
linguistic computation such that all inter-level mapping processes will be allowed for. The model will then tend to be non-derivational and will allow for inter-level mismatches in the presence of flexible interfaces. Contrarily, if a model of grammar possesses fewer sets of independent formation rules and primitives than levels of representation, some of the latter will be non-autonomous and some will be the computational source of all inter-level mapping processes such that at least some of the interfaces will be only uni-directional. The model will then tend to be derivational and it will tend to avoid inter-level mismatches in the search for transparent interfaces. However, I have stated that a hybrid approach to distributional and derivational features is possible if all levels of representation are relatively independent and show mismatches, yet some levels remain the computational source such that not all inter-level interfaces are allowed for.

**Direction**

The third architectural dimension studied in this work, Directionality, has been analyzed as the direction in which mapping processes take place between the various levels of a model of grammar. In this sense, Directionality refers to the direction in which interfaces are applied (from which levels into which levels), i.e. to the descriptive priority between the different levels of grammatical description (Zwicky 1972) or the direction which linguistic rules take in order to map structures through a grammatical system (Eliasson 1978: 50). I have used the term “top-down” to refer to those models in which the formalism begins with the conceptual intention and finishes in articulation (see Levelt 1989) (from meaning to form) and “bottom-up” for those models in which inter-level mapping processes take the opposite direction (from form to meaning).

I have also looked at Directionality as the direction of hierarchical relations between smaller and bigger units in an
intra-level fashion. The notion of Directionality is related to the direction of the mapping mechanism of a grammar and the intra-level, hierarchical relations that take place within the various strata. Here I have used the term “top-down” to refer to those models in which mapping takes place from bigger, root units into smaller units or leaves (from the topmost node down to terminal symbols) and “bottom-up” for those models in which smaller units are mapped into bigger ones (from terminal symbols up to a topmost node) (see Kimball 1973: 19-20).

I have further defined Directionality as the direction related to the goal of the grammar – whether a grammar seeks to generate language departing from abstract representations or to analyze language and convert it into abstract representations (see Newman 1990a, 1990b). This is related to a last notion of Directionality in which I have compared the Directionality supposedly involved in real-time processes and that implemented in inter- and intra-level mapping processes—whether a model of grammar is able to mimic the processes in which the speaker/hearer takes part when speaking/listening in the way in which the various levels of a grammar interact with each other and within themselves.

Accordingly, I have stated that Traditional Generative Grammar shows uni-directional inter-level Directionality in that the only directions allowed for in mapping processes are from syntax to phonology and from syntax to semantics. Inter-level Directionality in TGG does not therefore coincide with the direction involved in speech production (from meaning to sound via syntax) or in speech comprehension (from sound to meaning via syntax), since in the model syntax is at the beginning of any derivation. Intra-level Directionality is top-down (from the root, topmost node down to leave nodes, except for the MP, in which tree algorithms are built in a bottom-up fashion, see Chomsky 1993, 1995). Intra-level Directionality coincides therefore with the direction supposedly taking place in real-time parsing algorithms from bigger to smaller units (Kimball 1973) – except for the MP. As for the goal of the grammar, its very name
indicates that it is to generate language rather than to analyze it, although GG is most frequently used as a tool to analyze already produced sentences – thus it is used to reconstruct the derivational history of a given input string (see Longley & Stark 2001: 18).

Contrarily, the Parallel Architecture shows all-directional inter-level Directionality in that all directions are allowed for in mapping processes: from phonology to syntax and semantics; from syntax to phonology and semantics; from semantics to phonology and syntax. Inter-level Directionality in PA does therefore not coincide with the direction involved in speech production or comprehension. However, the levels are positioned such that syntax appears in the middle and semantics and phonology on the extremes, thus mimicking the direction involved in speech production and comprehension – and constraints in PA can be implemented in a specific order so as to fit specific processing tasks (Jackendoff 2010a: 5). Intra-level Directionality is such that sentences can be constructed either bottom-up or top-down without a particular algorithmic order (see unification in Shieber 1986). Although such intra-level Directionality goes against the claimed top-down direction supposedly involved in natural language parsing algorithms (Kimball 1973, see above), experimental evidence supports the all-directionality of incremental mapping processes (Ferreira 1996, Ferreira & Swets 2002, Jackendoff Fc.: 20, Levelt 1989, Wheeldon & Lahiri 1997). Lastly, regarding the goal of the grammar, PA is composed of three generative components and its goal is to generate language. However, as in GG, it is most frequently used as a tool to analyze already produced utterances.

Finally, Functional Discourse Grammar shows top-down inter-level Directionality in that function levels (pragmatics and semantics) map into formal levels (morphosyntax and phonology) (Hengeveld & Mackenzie 2008: 1-3). The model mimics therefore speech production, although it is a “model of encoded intentions and conceptualizations” rather than “a model of language production” and it “could in principle be turned on
its head to account for the parsing of utterances” (ibid: 2) such that it would be coherent with language comprehension. Intra-level Directionality in FDG is top-down, with parsing rules kicking in at higher layers first and lower ones only after (ibid: 125, 280, 311, 455), which is consistent with the posited direction of natural language parsing algorithms. Lastly, the goal of FDG is also to generate language. However, as with GG and PA, it is most frequently used as a tool to analyze already produced utterances.

I have further stated that there exists a close relation between derivational and directional features. A theory of language that shows derivational features will tend to show a directional approach. Contrarily, a theory of language that does not show clear inter-level directionality will not tend to show derivation in the processes that explain the product of language in the formalism. Accordingly, TGG, a strongly derivational framework, shows inter-level uni-Directionality – from syntax to phonology and from syntax to semantics. PA, on the contrary, is not a derivational framework and does not show inter-level Directionality either. FDG shows a certain degree of derivation and therefore inter-level Directionality – from function to form. I have also stated that there exists a preference for those models of grammar that have inter-level Directionality to also show intra-level Directionality and vice versa. Accordingly, both TGG and FDG show both inter- and intra-level Directionality while PA shows neither inter- nor intra-level Directionality.

The Domino effect

It has been confirmed that not all possible combinations of architectural features can take place when designing a grammar and that a domino effect takes place between the approach that a theory of language takes to the dimensions of Distribution, Derivationality and Directionality. In the default case scenario, a consistent architectural picture arises causing a domino effect of
architectural features all across the board. There seem to be two main possibilities. On the one hand, if distributional features are such that not all levels of representation possess their own primitives and formation rules, some levels of representation will necessarily be non-autonomous. This, in turn, means that there will be one or more levels from which all linguistic computation will need to begin such that not all levels of representation will be source-and-target but some will merely be source levels and some will merely be target levels. The model of grammar will then be derivational such that non-autonomous levels will be derived from that level or those levels from which all linguistic computation is born. Interfaces will then be transparent, since linguistic phenomena arguably taking place at one non-autonomous level of representation will have to be accounted for at the source level from which the former is derived. A model of these characteristics will in turn tend to be a directional model in which inter-level mapping processes are uni-directional –born from the deriving level and targeting derived levels. To my knowledge, in principle a model of grammar that conforms to this architecture will most probably show intra-level directionality between smaller and bigger units, though the opposite should also be possible. The scenario described above is illustrated by Generative Grammar.

On the other hand, if distributional features are such that all levels of representation possess their own primitives and formation rules, all levels will be autonomous. This, in turn, means that computation will start from any of the levels, since they will all be source-and-target levels –levels that may start off and receive computation. The model of grammar will then be non-derivational -no level will be completely derived from another level. Interfaces will then be flexible, since linguistic phenomena will not have to be mirrored at different levels of representation. A model of these characteristics will in turn tend to be a non-directional model in which inter-level mapping processes will be bi-directional. In keeping with the parallel nature of inter-level interfaces and all-directionality, intra-level
mapping processes will also tend to be parallel –computed either bottom-up or top-down. This is the case of the Parallel Architecture.

However, a violation of the domino effect between the 3 Ds may take place in that a single formalism may show features of Distribution, Derivationality and Directionality traditionally identified with opposing architectural tendencies. If distributional features are such that all levels of representation possess their own primitives and formation rules, all levels will in principle be autonomous. All levels will then be source-and-target levels. The model of grammar will then be non-derivational and interfaces will be flexible. Mismatches between the different levels will be allowed for in the theory. A model of these characteristics will in turn tend to be a non-directional model in which inter-level mapping processes will be bi-directional. Nonetheless, a model of grammar may have independent formation rules and primitives for all levels of representation and still be strongly directional such that derivation needs to start from one or more levels of representation. The scenario described above is illustrated by the architecture of Functional Discourse Grammar.

Adequacy

It has not been the goal of this thesis to determine which theory is superior from any standpoint, but rather to establish a series of parameters according to which the architecture of a theory of grammar can be analyzed and further features can be predicted, as well as to be able to relate the former to the use of a particular type of theory-internal devices. Different theories of language have different goals, and it is according to these goals that they ought to be judged.

If a theory of language wishes to account for linguistic “competence” (an ideal speaker/listener’s knowledge of language unaffected by memory, concentration limitations,
speech errors, etc, see Chomsky 1965: 3) such that language acquisition can be explained, my assumption is that a theoretical system can be as idealized as one wishes, since it only has to be consistent for and within itself—a self-contained construct. From this point of view, I believe no model of grammar can, as of today, reach the analytical complexity that the generative framework has reached in its analysis of the syntactic component—then again, the efforts of the generative community have been rather focused upon that one component, with the evident outcome that syntax has been developed in that community more than anywhere else. If the generative framework is designed to merely account for idealized competence, I see no real problems in the architecture it displays: its possibly incomplete distribution (no account for pragmatic inferences—language use out of its reach); its derivational design, in which syntax appears as the main computational component (against both speech production and comprehension, which the framework does not seek to mimic in any case); its reduced directionality (no interaction between semantics and phonology, or impact of semantics and phonology upon syntax); its non-incremental mechanism (sequenced steps need to be fully finished before proceeding on to the next step); its use of empty categories with no corresponding perceivable structure or impact on the surrounding perceivable structure—empty sets are actually one of the main (non)discoveries in mathematical theory.

If a theory of language, however, wishes to be coherent with linguistic “performance” (the production of actual utterances) as well as with linguistic “competence”, with evolutionary biology and other cognitive sciences, then the architecture of the model needs to be consistent with these—and with what is actually perceivable in linguistic utterances. In order to account for more than idealized linguistic competence, the Parallel Architecture adopts a different architecture of grammar: a non-derivational design in which syntax loses its ruling position (Burling 2003); more comprehensive
directionality (impact of semantics and phonology upon syntax, and interaction between semantics and phonology -Jackendoff 2002: 125); an incremental mapping mechanism (sequenced steps do not need to be fully finished before proceeding on to the next step but rather information from one component is passed on to another one as soon as enough information is available –Jackendoff 2002, 2007, fc.); reduced use of empty categories (traces are the only possible empty elements); the simpler syntax (Culicover & Jackendoff 2005) that results from this architectural configuration. As to its distribution, the lack of a pragmatic component forces the conceptual structure to take on board both semantic and pragmatic features, or else consider the latter to be non-grammatical.

Finally, Functional Discourse Grammar puts emphasis on “performance” and on the way in which the grammar reflects language use, which is clear from the fact that its distribution includes a pragmatic level -in fact, the interpersonal level is the main level, from which all linguistic computation must begin. Although the model is not intended as “a model of language production” (Hengeveld & Mackenzie 2008: 2), its directionality from function to form is coherent with that of language production. Also consistent with natural language processes is its incremental mapping between levels (Depth First Principle, ibid: 24). These distributional and directional features lead, however, to the following: more restricted directionality (strongly top-down, with limited access from formal to function levels); reduced use of empty categories (e.g. in raising and control, although vacuous representations are kept to a minimum, Maximal Depth Principle ibid: 25).

**What if the glass is empty?**

A linguistic school’s belief in, and account of, empty categories depends upon the main architectural tenets of the theory at hand. The licensing conditions of empty categories are a direct
consequence of various factors defining the organization of a theory of language: Are the various levels of representation independent or derived from each other? Is there a clear priority of one level of representation over other levels? Does the theory of language allow for mismatches in the quantity and type of representations that appear at the different levels of representation or does it search for a fully transparent interface between them?

Empty categories may arise in non-default mappings between meaning and form, in violations of syntax-semantics interface transparency. Since there is an apparent mismatch between form and function, it is interesting to examine whether a theory of language accounts for such a mismatch by means of a discrepancy in the representation at various levels, or whether some kind of empty category is introduced to avoid such a mismatch. This choice, in turn, depends on the whole architecture of the theory.

Syntax-semantics transparency can be violated in several ways. A first way to contravene syntax-semantics transparency is shown by linguistic phenomena in which function seems to be richer than form, i.e. more is meant and understood than is actually said. This type of infringement upon the transparent syntax-semantics interface can be argued to take place a.o. in the following phenomena: understood arguments (Ackema 2002, Fillmore 1986, García Velasco & Portero Muñoz 2002, Groefsema 1995, Han 2000, Van der Wurff 2007); sluicing (Brucart & MacDonald 2012, Merchant 2001, 2003, 2011); small clauses (Aarts 1992, Chomsky 1981, Huddleston & Pullum 2002, Lundin 2002, Quirk et al 1985, Williams 1974, 1983); and interjections (Ameka 1992, Bennis 2006, Greenbaum et al 1985, Matamala 2007, O’Connell, Kowal & Ageneau 2005, Pompino-Marschall 2004, Wharton 2000). I have named this type of mismatch a quantitative mismatch. In these cases, a theory of language may represent more categories in semantics/pragmatics than in syntax, thus creating a representational mismatch in the quantity and length of representations for function and form. Alternatively, a
A theory of language can avoid the form-function mismatch by introducing an empty, unpronounced syntactic category at syntax to provide those semantic categories lacking an overt syntactic counterpart with a syntactic counterpart.

A further type of violation of the default transparent syntax-semantics interface is illustrated by raising (Hornstein 1999, Polinsky & Postdam 2006), in which the linearity of meant and understood linguistic material does not correspond with the linearity of syntactic material. In raising, an argument in an embedded clause corresponds to a syntactic phrase in the matrix clause. This I have called a qualitative mismatch of the distributional type. In raising, a theory of language can opt to represent the overt linearity of what is said at syntax and leave for semantics the order in which those syntactic constituents are meant or understood, thus creating a mismatch in the distribution of representations. A theory of language can alternatively introduce an empty category at syntax, that where the semantic argument “occurs”, such that the mismatch is avoided.

I have further used control (Brainard 1997, Hornstein 1999, fe., O’Neil 1995a, 1995b, Rooryck 1992) to illustrate a violation of both the quantitative and the distributional type. In control, one syntactic element corresponds to two different semantic arguments - meant and understood linguistic material seems to be richer than syntax (creating a mismatch of the quantitative type) while the linearity between them is infringed upon (creating a mismatch of the distributional type). Thus, a theory of language may choose to maintain both mismatches by representing one category at syntax and two at semantics (only one of which has a location corresponding to that of the syntactic category) such that both quantity and distributional mismatches are reflected in the formalism. Contrarily, a theory may choose to avoid both mismatches by introducing an empty category at syntax such that both semantic arguments have a syntactic counterpart and syntactic linearity corresponds to the semantic one.

Finally, I have discussed a further type of violation of the
transparent syntax-semantic interface of the qualitative type illustrated by pseudo-coordination (Culicover 1970, 1972, 2010, Culicover & Jackendoff 1997, Kwon 2004) and pseudo-subordination (Yuasa & Sadock 2003, see also De Vries 2002, 2006, Jackendoff 2010a). In pseudo-coordination and pseudo-subordination, semantics and syntax do not differ in quantity or distribution. Rather, they contravene the default association that normally takes place between semantic and syntactic categories. In pseudo-coordination, syntactic coordination is used to express subordination. In pseudo-subordination, syntactic subordination is used to express coordination. A theory of language may choose a symmetric relation for syntax and an asymmetric relation for semantics in pseudo-coordination and the opposite case scenario for pseudo-subordination, thus creating a mismatch. Alternatively, a theory of language may avoid such a mismatch by arguing that the non-default subordinate and coordinate readings arising in pseudo-coordination and pseudo-subordination respectively are the product of non-grammatical inferences.

Some conclusions can be drawn as to the main architectural tenets of a linguistic theory that chooses to follow one or the other path to account for the phenomena illustrated above and, in general, any phenomenon that triggers some kind of discrepancy between function and form. A model of grammar that reflects these discrepancies in the formalism is necessarily a model of grammar that allows for a flexible interface because meaning and form levels are independent from each other – neither is the semantic level derived from the syntactic one, nor is the syntactic level derived from the semantic one. Such architecture of grammar avoids the use of empty categories to the detriment of representational transparency. This relation has been stated in the workings of the Parallel Architecture.

The opposite case scenario arises if a model of grammar does not reflect these discrepancies between function and form in the formalism. A theory that does not allow for representational mismatches is a theory in which meaning and form levels are dependent from each other (one is derived from
the other) such that the transparent interface can be maintained. This leads to the introduction of empty categories in order to provide the semantic representation of various phenomena with a syntactic counterpart. This is a logical move if semantics is interpreted from syntax. Such architecture of grammar avoids representational mismatches at the cost of proximity to the actual perceivable structure. This relation has been confirmed in the workings of Traditional Generative Grammar.

The relation between the use of empty categories and the architectural tenets of a theory of language as defined by the 3 Ds is however best illustrated by Functional Discourse Grammar. In this framework, empty categories are avoided in an attempt to keep representations at formal levels (morphosyntax, phonology) as close as possible to the perceivable structure, thus making use of representational mismatches. Although against the general tendency of the model, raising and control are accounted for by introducing empty categories at the morphosyntactic level of representation that provide a formal counterpart for the functional categories introduced at Formulation, thus avoiding a discrepancy between representations at different levels. An architecture of grammar that follows an inconsistent path regarding the use of mismatches and empty formal categories is one that goes against the Domino effect which usually takes place between the 3 Ds. Functional Discourse Grammar shows contrasting distributional, derivational and directional features. On the one hand, all levels possess distinctive primitives and formation rules, thus they contain specific information and are able to show mismatches between them. On the other hand, the levels’ inter-independence is constrained by the model’s strong directionality— all computation must necessarily start at function levels such that there is a sequence in the implementation of the model and, from this perspective, formal levels are partly derived from function levels. The inconsistency in the choice of distributional, derivational and directional features shows that the domino effect can be infringed upon, and that the architectural
inconsistency is passed on as a mixed use of empty categories and mismatches.

There is thus a close relation between the various features that constitute the architecture of a grammar: the number and type of its levels of linguistic representation; the relations among them; the priority of certain levels upon others. The three architectural Dimensions studied in this book (the Distribution of information into various strata, the Derivation of one level or sub-level into another, the Direction of interfaces) cover these features such that a 3D analysis of a model of language results in a comprehensive description of a given theory with which further architectural features can be predicted. These architectural parameters, in turn, predict the use of empty categories or of representational mismatches in a given theory of language when representing particular phenomena that seem to violate the default transparency of the function-form interface. While empty categories maintain inter-level transparency by representing assumed absent linguistic elements in the search of one-to-one mappings between possibly redundant levels, mismatches in the representations of the various levels allow a model to remain closer to perceivably present linguistic elements by violating interface isomorphism. Various theories of language solve infringements upon language transparency in different ways - either by “complicating” one level with the introduction of an empty category, or by “complicating” the interaction between various levels with non-homomorphisms. The question remains then how to solve the puzzle: by reconstructing assumed missing pieces, or by claiming that those pieces were never there in the first place, and that the puzzle actually looks totally different?