To appear in Keith Brown (Ed.): *Encyclopedia of Language and Linguistics, Second Edition*. Oxford: Elsevier.

Declarative models of syntax

Robert D. Levine and W. Detmar Meurers
The Ohio State University

1 Introduction

Theoretical linguistics—which from its earliest days has self-consciously modeled its working methodology on that of the natural sciences—continues to face a dilemma of long standing which interestingly does not seem to be an issue in the established empirical disciplines. On the one hand, parallel to the situation in other scientific domains, the investigation of the conditions characterizing well-formed human language is logically independent of the question of how a procedure can be implemented that produces or licenses grammatical utterances. Yet, at the same time, early generative syntax and more recent research in the wake of Chomsky (1995) does not separate the meaning of a theory from its procedural, computational realization. Instead, the theory itself is directly tied to the mechanism by which a given string is procedurally derived.

With linguistic generalizations directly tied to procedural mechanisms, progress in generative linguistics arguably has been hampered by ordering paradoxes and other theory-internal issues, and the procedural mechanisms or metaphors have stood in the way of more empirically adequate, declarative generalizations (cf. section 2)—deficiencies which have led to the development of declarative linguistic frameworks.

Declarative approaches to syntax, such as Head-Driven Phrase Structure Grammar (HPSG, Pollard and Sag, 1987), Lexical-Functional Grammar (LFG, Bresnan, 1982) and others (cf., Borsley and Börjars, to appear), follow the otherwise standard scientific practice of seeking a declarative characterization of the research domain and its properties, i.e., they investigate the constraints which are observed to hold of linguistic representations. Declarative models of syntax are not committed to any particular processing model or procedural mechanisms for deriving linguistic representations.

Representations Declarative models of syntax are commonly identified with monostratal representations, i.e., the use of a single level of representation. This association requires some discussion since analyses in declarative frameworks do appear to make use of multiple levels of representation, at least until it is clarified what exactly is meant by a level of representation.

Within the HPSG paradigm, for example, Bouma et al. (2001) propose a declarative approach to unbounded dependency constructions which is based on principles relating the argument structure (encoded under an attribute called ARG-ST), the dependents list (DEPS), and the valence attributes (SUBJ, COMPS). This essentially establishes three levels of representation, with the argument structure encoding the interface to the semantic arguments of a predicate and the valence attributes encoding which of these are realized how in the sentence. But crucially, all three representations are explicitly and transparently present in the model of the linguistic objects being licensed, i.e., they are a part of what procedural architectures often refer to as the syntactic surface structure. In other words, declarative approaches are monostratal in that all representations which play a role in licensing a particular string are simultaneously and completely part of the model of the linguistic object being licensed.

Compare this to the situation in a derivational paradigm, for which, e.g., Chomsky (1995, ch. 4) states that "computation typically involves simple steps expressible in terms of natural relations and properties, with the context that makes them natural 'wiped out' by later operations, hence not visible in the representations to which the derivation converges." In other words, in place of the constraints on a single, transparent surface structure representation assumed in a monostratal, declarative approach, the derivational architecture relies on multiple levels which are connected by procedures that "wipe out" evidence for their nature on the way to the observable surface representation. The derivational approach thus assumes hidden levels of representation and formulates grammatical principles on this sequence of procedurally related, underlying representational layers.

In sum, a key characteristic of a declarative approach to syntax is that it involves an explicit data structure in which all information is simultaneously represented, i.e., the surface structure object that is licensed by the theory includes all information that plays a role in licensing it.

Locality and Modularity The nature of the representations assumed in declarative and procedural architectures directly relates to another important issue: the nature of the domains in which grammatical relations are established.

In classical transformational grammar (TG, Chomsky, 1965), transformations can in principle apply to any of the nodes in the tree and have any effect; i.e., the grammatical relationships they establish can involve any of the elements in the domain of the entire tree. In the more recent derivational incarnation, Chomsky (1995) holds that "in syntax, crucial relations are typically local, but a sequence of operations may yield a representation in which the locality is obscured." Common to the derivational approaches thus is that the model of the surface structure does not include direct evidence of the locality of grammatical relations.

This contrasts with declarative models of syntax, in particular in the HPSG paradigm, where constraints are generally formulated for local trees, and non-local relations are established by locally recording the non-local relationship. This local recording is empirically well-motivated given the extensive and robust documentation of languages in the Indo-European, Bantu and Austronesian families exemplifying the local recording of nonlocal connectivity through a diverse spectrum of morphological, syntactic and phonological flags (cf., e.g., Zaenen, 1983; Hukari and Levine, 1995).

Since all relationships are locally recorded in the model of the surface structure, they form independent, exchangeable modules of linguistic expressions. Declarative syntax thus is modular in the sense that the entire syntactic representation results from the combination of local, modular representations.

Relevance of the declarative/procedural distinction The discussion of the declarative/procedural framework dichotomy at times seems to have a rather abstract quality, as though issues of analytic practice and the formulation of satisfactory hypotheses about particular phenomena could always be encoded equally straightforwardly in either approach. Movement and feature-matching, for example, might be viewed as nothing more than different formulations of the notion of structural involvement, whereby two nodes in a structure are implicated in a dependency of some kind.

While hypotheses about linguistic structure and generalizations can indeed embody core insights which are portable from framework to framework, this does not change the fact that frameworks differ with respect to which language properties are modeled, how linguistic relations can be established, and how generalizations can be expressed. They both influence and restrict the kind of hypotheses that *can* be supplied, and, as a result, a solution which involves a natural application of one framework's resources may be impossible or extremely awkward to simulate in a different framework.

Before we turn to illustrating the empirical and conceptual differences arising from the choice of a procedural vs. a declarative framework in section 2, it is important to mention that the choice of a procedural or a declarative model also has clear formal consequences. McCawley (1968) showed that context-free grammars can be interpreted in two distinct ways: as string/tree rewriting systems or as node admissibility conditions. The application of rules in a rewriting systems are ordered, which gives rise to ordering paradoxes and questions of extrinsic ordering of rules—two issues we return to in section 2.4. These issues do not arise when interpreting the rules as node admissibility conditions; a tree is admitted iff all its nodes are admitted, so there is a logical conjunction of admissibility conditions, for which order is irrelevant. Interestingly, Peters and Ritchie (1973) show that a generalization of node admissibility to context-sensitive rules still only licenses context-free languages (cf., also, Perrault, 1984; Jacobson and Johnson, 1991; Oehrle, 2000). The formal question of the interpretation of a grammar in terms of rewriting or node admissibility thus has clear formal consequences in addition to the empirical/conceptual ones we are focusing on in this article (cf., also, Johnson, 1994; Pullum and Scholz, 2001).

Declarative explanations as the standard in the natural sciences As background for the declarative/procedural distinction in syntactic theorizing, it is instructive to consider the situation in other empirical sciences. Whether one is attempting to account for the behavior of matter and energy, the behavior and structure of land forms or life forms, or the large-scale organization and history of the universe, the form that explanatory theories take invariably is a set of conditions that must be met among certain fundamental elements; the world we observe is just the world that conforms to these conditions. All such conditions consist in equalities (such as the conservation principles in physics) or inequalities (such as the uncertainty principle which holds of all non-commuting operators corresponding to dynamical variables in quantum mechanics). Scientific theories of the natural world in general do not posit a series of procedures or quasi-events *as part of the constraint system constituting the explanatory hypothesis*, where the outcome of the procedure defines admissibility for a possible description of the system and therefore for some set of events in the world we are observing (cf., Johnson and Lappin, 1999, ch. 5, for a closely related discussion).

2 Empirical and analytic arguments

This section fills the claim with life that the differences in the linguistic architectures have direct consequences on the analyses which can be proposed in them. The advantage of declarative approaches to syntax become apparent in the analysis of phenomena in which grammatical relationships are established that go beyond the one-to-one correspondences readily established by movement in procedural approaches to syntax.

We discuss four classes of issues which showcase the advantage declarative approaches have over procedural ones: i) multiple gap constructions, where several gaps are satisfied by a single filler, ii) phenomena where a single argument requirement is realized by two constituents, iii) partial constituent phenomena, and iv) the spurious problems resulting from procedural approaches by way of ordering paradoxes and multiple derivations with the same outcome.

2.1 Multiple gap constructions

The key procedural component to transformational grammar is, of course, the set of transformations themselves. Transformations are procedural in a particular sense which can be clearly illustrated by movement rules. Consider the relationship holding between a filler and a gap site in a sentence such as (1).

(1) **Who** do you think Robin believes the police will arrest __?

In transformational grammar, such a sentence is licensed by a series of steps, illustrated in (2).

(2) a. You think the police will arrest **who**.
b. You think Robin believes **who** the police will arrest __.
c. You think **who** Robin believes __ the police will arrest __.
d. **Who** you think __ Robin believes __ the police will arrest __.

There is a series of steps which must be carried out in a particular order in order to ensure that the various locality conditions which have been such a conspicuous part of linguistic theorizing since the early 1970s (cf., e.g., Chomsky, 1973, 1977, 1981) are respected. A movement of the filler *who* must result in its appearance in some left-peripheral position attached to the lowest clause before it moves to the next higher clause, and so on. Movement directly from the original position to the periphery of the second clause from the bottom is blocked, on one or another formulation of these locality requirements. Thus the extraction is licensed in separate steps, which must occur in a particular sequence.

It has sometimes been suggested that this mechanism could be replaced by a purely 'base-generated' account of filler-gap linkages, where empty categories were freely distributed and linked to overt fillers and each other, by co-indexation and declarative conditions on chains; Chomsky (1981, pp. 89–92) raises this possibility, and essentially dismisses it as an uninteresting variation of the basic movement mechanism assumed from the earliest period of transformational grammar. But this view is seriously mistaken; the procedural treatment of extraction by a series of steps which must occur in a particular order is not equivalent to the free generation of empty categories in particular locations which must satisfy essentially the same conditions on the legal form of $\bar{\text{A}}$ chains. The discrepancy is not evident when links between an overt gap and a single filler are considered. But the picture is quite different when that filler is linked to an arbitrary number of gap sites, as it is in the case of across-the-board (ATB) coordination extractions, parasitic gaps, and related multiple-gap phenomena.

Parasitic gaps Consider examples such as

(3) a. Who do you think the police will arrest after interrogating?
  b. Who do you think the police will arrest and the prosecutor will indict for this crime?

Allowing empty categories in the base representation to remain unfilled with lexical content, and to freely co-index with the filler, yields legal representation of the form

(4) $\textbf{Who}_i$ you think $\_\__i$ the police will arrest $\_\__i$ after interrogating $\_\__i$.

The procedural approach, however, must start with the representation in (5).

(5) The police will arrest $\textbf{who}_i$ after interrogating $\boldsymbol{NP}_i$.

The valence requirements of *interrogate* demand an underlying NP in object position, which must somehow correspond to a gap linked to the filler. But this solution cannot simply replicate the ordinary single filler-gap connectivity mechanism assumed for (1). The problem, as emphasized in Gazdar et al. (1982), is that the procedure implementing the transformation-based relation between filler and gap site, *wh*-movement, has never been defined for more than a single *pair* of positions in adjacent stages of a complete derivation. In the absence of any definition of the notion 'transformation' encompassing multiple targets of a single movement operation, the nature of the NP in (5) has to be interpreted as something other than a copy of the overt *wh* filler. There have been a number of proposals based on this strategy, each of which is burdened with serious conceptual and/or empirical weaknesses.

Chomsky (1982) takes NP in (5) to be *pro*, a null pronoun which is interpreted at Logical Form (LF) as a variable, in accordance with his theory of mutable status for empty categories—a position he abandoned within a few years;

Chomsky (1986) adopts Contreras's (1984) suggestion that parasitic gaps be regarded as the traces of extraction, entailing the existence of a movable covert filler at the parasitic gap site and a special relationship called 'Chain Composition' linking the true and parasitic chains in a manner consistent with the rest of the theory— an explicitly construction-specific treatment in the grammar along lines which Chomsky (1982) had heaped scorn on.

Frampton's (1990) treatment of parasitic gaps, a kind of hybrid of Kayne's (1994) connectedness with the null operator treatment in Chomsky (1986), incorporating the *wh*-deletion analysis of Chomsky (1977), is in effect a derivational reconstruction of the multiple licensing of extractions pathways linked to a single filler. Serious empirical difficulties with this proposal are discussed in Levine and Sag (2003) and Levine (2004), including the conspicuous irrelevance of certain data first noted by Kearney (1983), cited in both Chomsky (1986) and Frampton (1990), which seem to distinguish the reconstruction behavior of parasitic chains from those of true chains, an issue which becomes relevant in the discussion of ATB extraction below.

A different approach to the problem is offered in Cinque (1990), who takes parasitic gaps to reflect the presence of a null resumptive pronoun with an Ā-position binder. The empirical basis for this proposal and related proposals has been extensively contra-exemplified; see Levine et al. (2000), Levine (2001). Thus the key empirical foundations of such analyses are unsupported.

It will be seen from this brief summary that the primary difficulty of procedural solutions to multiple-gap extractions is in fact their reliance on a procedure—movement—which is undefined for more than a single gap. A declarative approach, however, faces no such problem, since multiple linkages via feature-matching (as in Gazdar 1981, Gazdar et al. 1985, and Pollard and Sag 1994) are possible and involve machinery no different from what is required for single filler-gap linkages. Similar observations apply to ATB extraction phenomena, which we turn to next.

ATB extraction In emphasizing the importance of the already mentioned Kearney paradigm, Chomsky (1986) called attention to the putative separation those data induce between parasitic gap phenomena and what he refers to as 'multiple gap constructions', without specifying the latter; it seems evident, however, that such constructions can be nothing other than ATB coordinate extractions. But when one tries to identify a robust transformational analysis of ATB extraction, the situation is no better, for reasons precisely parallel to the parasitic gap case: a single application of the TG movement rule cannot affect more than a single target and a single destination site. Thus, excluding theoretically eccentric positions such as Goodall's (1987) three-dimensional syntactic space solution, there seems no alternative to treating ATB extraction itself in terms of empty operator movement, and this is exactly the solution proposed in Munn (2001), which is virtually identical to Chomsky's (1986) treatment of parasitic gaps. The empirical soundness of this has however been sharply questioned in Borsley (1994), and Munn's approach gives every appearance of being motivated solely by the need to reconcile the existence of multiple gaps linked to a single overt filler with the limitation inherent in a transformational treatment of extraction. We note in passing that while Munn has argued for a parallel treatment of parasitic gaps and ATB extraction as justified by their shared immunity to weak-crossover (WCO) effects, the WCO facts provide no support for the analysis that Munn proposes, viz., null pronominal interpretations at LF; see Levine and Hukari (to appear) and also Postal (1993b) for discussion of the empirical issues involved.

2.2 Multiply realized requirements

Conceptualizing linguistic relations in the procedural terms of movement has the consequence that an entity missing in one place is expected to be realized in exactly one other place, the one it has been moved to. As we saw in the previous section, this assumption is inadequate for phenomena involving a single constituent being linked to multiple gap sites. The second, related possibility is that a single requirement could be realized by multiple constituents—and indeed, declarative approaches to phenomena such as the Split-NP construction in German have shown that this possibility is well-worth exploring.

The German Split-NP construction, exemplified in (6), received some attention in the 80s since the direct object seems to be split across two locations, the noun *Schokoeier* is fronted while the determiner *drei* occurs in the base position (often referred to as the *Mittelfeld*).

(6) Schokoeier fand er drei.
chocolate eggs found he three
'As for chocolate eggs, he found three.'

Van Riemsdijk (1989) provides an account for this phenomenon in which the fronted element (e.g., *Schokoeier*) has been moved to this position, leaving some material (e.g., *drei*) behind. However, such a movement-based account of the construction runs into a significant obstacle in light of examples such as van Riemsdijk's (7).

(7) Ein Schwimmbad hat er sich noch keins/*kein gebaut.
a pool has he REFL *yet none/no built*
'As for a swimming pool, he hasn't yet built one.'

(8) a. Er hat sich keins gebaut.
he has refl none built

b. Er hat sich ein Schwimmbad gebaut.
he has refl a pool built

The example (7) is of interest since the fronted *ein Schwimmbad* and the *none* in the *Mittelfeld* appear to be two complete constituents. Even if one ignores the presence of the indefinite determiner *ein*, combining the fronted material with the supposed remnant in the *Mittelfeld* would result in *kein Schwimmbad*, i.e., a form of *kein* which is different from the one obligatorily showing up in (7).

The fact that both NPs behave like two autonomous NPs is entirely unexpected under a movement-based approach and Van Riemsdijk resorts to positing that non-maximal projections that have been moved can under certain conditions "regenerate" structure to meet X-bar requirements at S-structure. However, even this deus-ex-machina cannot save the movement-based analysis in light of examples where the NP is fronted as part of a verbal projection, as is (9) taken from Kuhn (2001).

(9) [Einen Wagen gekauft] hat er sich noch keinen.
a car bought has he REFL *yet none*

'As for cars, he has not yet bought one.'

In conclusion, it is empirically inadequate to rely on movement to procedurally establish a relation between the two constituents in the German Split-NP construction. A declarative analysis licensing both NPs independently does not suffer from the shortcomings of a movement-based account since it is not restricted to establishing such one-to-one relation. The properties of the Split-NP construction are explained without recourse to movement, which has been worked out in terms of so-called "base generation" approaches in the transformational framework (e.g., Fanselow, 1988; van Geenhoven, 1998) as well as by approaches in strictly declarative framework (e.g., Kuhn, 2001).

2.3 Partial constituent phenomena

The occurrence of partial constituents in German has received much attention in the literature since the phenomenon is a serious challenge to a theory of constituency. The data in (10), taken from De Kuthy and Meurers (2001), illustrate the phenomenon.

(10) a. [Verkaufen] wird er **das Pferd**.
sell will he the horse

'He will sell the horse.'

b. [Ein Buch] hat Hans **über Syntax** ausgeliehen.
a book has Hans on syntax borrowed

'Hans borrowed a book on syntax.'

c. [Stolz] ist er **auf seine Kinder** gewesen.
proud is he of his children been

'He was proud of his children.'

In (10a), the transitive verb *verkaufen* has been fronted, leaving its complement, the NP *das Pferd*, behind. In (10b), the NP *ein Buch* is topicalized without its PP complement *über Syntax*. And in (10c), the AP *stolz* has been fronted, leaving its PP argument *auf seine Kinder* behind.

The two analysis ideas that have been proposed in the literature are that (i) the constructions involve movement of a constituent containing a trace, so-called *remnant movement* (Thiersch, 1985, Webelhuth and den Besten, 1987, G. Müller, 1996) or that (ii) the 'small' constituents which seem to occur in them are taken at face value and licensed by a *reanalysis* process (Fanselow, 1987, for partial NPs).

The in principle attractive idea behind the remnant-movement account is to reduce certain patterns of ungrammaticality to general restrictions on movement, such as supposed subject-object asymmetries, specificity and freezing effects. But as De Kuthy and Meurers (2001) discuss in detail, there are counterexamples to each of the supposed restrictions and the general reduction of displacement to movement cannot explain the contrast between scrambled and fronted constituents illustrated by (11) and (12).

(11) a. * Er wird [das Pferd verkaufen] noch heute wollen.
he will the horse sell still today want to

'He will want to sell the horse today.'

b. [Das Pferd verkaufen] wird er noch heute wollen.
the horse sell will he still today want to

(12) a. * Er wird [verkaufen] **das Pferd** noch heute.
he will sell the horse still today

b. [Verkaufen] wird er **das Pferd** noch heute.
sell will he the horse still today

Interestingly, one of the arguments against reanalysis fielded by G. Müller (1991, p. 175) is that a reanalysis rule cannot properly be expressed in a syntactic framework which relies on movement as its key explanatory device. The assumption of a derivational architecture thus turns into a straight jacket, ruling out the formulation of a potentially sensible analysis.

This contrasts with a declarative approach to syntax supporting constraints that go beyond the one-to-one relations established by movement. For the empirical domain at hand, De Kuthy and Meurers (2001) show that a reanalysis-like approach can be formulated and made concrete in the form of declarative constraints on the explicit linguistic data structures assumed in the HPSG framework.

2.4 Self-created problems of procedural models

By virtue of defining its licensing criteria as procedures, the transformational approach requires them to interact sequentially. The result is that grammars with a procedural architecture have difficulties built into them which are purely artifacts of the ordering possibilities associated with that architecture. Some illuminating examples of this problem are afforded by the very early history of transformational grammar, in which not only transformations but the rules of the base component were taken to be necessarily ordered. So, for example, Bach (1964) envisages a grammar for an imaginary language containing a VP expansion rule

(13) $\text{VP} \rightarrow \left\{ \begin{array}{c} \text{Verb}_i \\ \text{NP} + \text{Verb}_t \end{array} \right\}$

and a rule for 'nominals'

(14) $\text{Nom} \rightarrow \text{Noun} \left\{ \begin{array}{c} \text{Sg} \\ \text{Pl} \end{array} \right\}$

Bach (1964, pp. 45–46) then observes that

> the rules are arranged in such a way that they may be applied one after the other without the necessity of returning to an earlier rule, although a single rule may be applied several times when necessary. For example, the verb phrase (VP), which may contain further instances of the noun, is developed before the latter is expanded. In general, rules that apply to a class of items should be placed before the rules that split the class into smaller subclasses, since if the other order is followed the general rule can be applied only naming each of the subclasses involved.

But if each of the rules simply identifies a class of (sub)trees which are to be regarded as legal objects, with other structures not explicitly sanctioned by rule defined as inadmissible, then clearly the order in which the NP and VP expansion rules are encountered is thoroughly irrelevant, making as little difference to the outcome as the order in which a human arithmetician combines pairs of numbers out of a large sequence of numbers to be added up in order to come up with a result. Conversely, a machine, programmed to rewrite symbols on the basis of further information about those symbols, would of course find it impossible to proceed if a symbol only introduced by a later rewrite operation was required before that operation had taken place. On a declarative interpretation of a set of phrase structure rules, where each rule does nothing more than specify a class of admissible structures, it is irrelevant to the denotation of the grammar—the class of objects admitted—in what order the well-formedness conditions are checked.

But it is not solely phrase structure rules which (under their early procedural interpretation) created purely internal problems which necessitated the introduction of otherwise unmotivated devices, such as rule ordering, in syntactic theory. Transformations themselves, the procedural technology par excellence that generative grammar made so theoretically central nearly half a century ago, create their own difficulties and assumption-bound contradictions, requiring solutions whose status is completely tied to that of the procedural model which in effect created them. A persistent example is the often-referred to but never formulated 'recovery of deletion' principle (Matthews, 1961, cf., e.g.,); a much more recent instance is the notion of 'economy of derivation' (Chomsky, 1995, cf., e.g.,). We illustrate our point here with the notion of the cycle, introduced to overcome a problem arising from the interaction among externally ordered rules.

A concise description of some of the problems such interaction causes is given in Culicover (1976). Culicover notes examples such as

(15) a. Mary expects John to visit George.
b. John is expected by Mary to visit George.
c. Mary expects George to be be visited by John.
d. George is expected by Mary to be visited by John.

and observes that examples (b)-(d) appear to require a contradictory ordering arrangement: in (b), Passive must precede Raising (or *John* will not become a subject and hence will not Raise; but in (c) *George* can only have been passivized *after* it Raised, etc. He explains that

> The reason why this is not a contradiction of linear ordering is that ... it appears to be possible to maintain the linear ordering of transformations if the transformations are applied in sequence first to the lowest S in the tree, then to the next highest, and so on. This method of applying transformations is called the Cyclic Principle ... the linear ordering [Raising $\succ$ Agent Postposing $\succ$ NP Preposing can be maintained if the Cyclic Principle is adopted. Without this principle, the data constitute a strong counterexample to the principle of linear ordering.

The last sentence in this passage is the crucial point in terms of our argument, because it can be rephrased as follows: the Cyclic Principle is *justified* by the consequence that linear order among transformational steps can be maintained. But the existence of linear order is itself an entailment of a procedural architecture. Thus, the import of Culicover's discussion can be fairly summarized by the observation that the existence of the cycle is a question which only arises because of the assumption that the theory of grammar be couched in procedural terms, the application of structure-to-structure mapping rules whose joint generative yield depends on the order in which the procedures corresponding to each rule are carried out. Under a declarative architecture, the notion 'order of application' makes no sense, and therefore the notion of a 'cycle' is otiose, or more correctly, formally meaningless.

3 Summary

Generative syntax was originally developed based on phrase structure trees as the exclusive syntactic data structure, which made derivations transforming one tree into another the key mechanism for expressing syntactic relations. The procedural nature of the application of rules and transformations leads to ordering paradoxes and other theory-internal complications. And the use of movement as a transformation relating two tree positions expresses one-to-one relations, which is empirically inadequate as a means to capture the one-to-many, many-to-one relations and the cases of flexible constituency exhibited in natural language.

Declarative models of syntax were developed to overcome these shortcomings by providing—in line with standard practice in the natural sciences—a logical, non-procedural characterization of the empirical domain. They are based on an inventory of more general linguistic data structures, such as (typed) feature structures, which make it possible to explicitly model the full inventory of linguistic properties, and they support a wider range of constraints expressing linguistic relations and generalizations. Since declarative models support an explicit formalization of linguistic relations and generalizations that go beyond one-to-one relations, they can provide empirically adequate theories for a wider range of data. The rewriting steps and transformations are replaced with conditions that are required to hold of tree nodes, and, more generally, the linguistic objects modeled. The problematic sequencing of procedures thus is replaced with a logical conjunction of conditions which have to hold. Declarative models of syntax realize the scientific architecture typical for the natural sciences by separating the modeling of the domain and the logical characterization of the generalizations which hold in it from the design of algorithms which procedurally realize such models. Complementing the success of declarative models of syntax in the empirical domain, this has also made such models the first choice for natural language processing, where a wide range of algorithms, from parsers and generators to more general constraint resolvers, have been developed to process with them.

References

Bach, E. (1964). *An introduction to transformational grammars*. New York: Holt, Rinehart and Winston.

Borsley, R. (1994). In defense of coordinate structures. *Linguistic Analysis* 24, 230–231.

Borsley, R. D. and Börjars, K. (eds.) (to appear). *Non-transformational syntax: A guide to current debate*. Oxford, UK: Blackwell.

Bouma, G., Malouf, R. and Sag, I. A. (2001). Satisfying constraints on extraction and adjunction. *Natural Language and Linguistic Theory* 19(1), 1–65. ftp://csli-ftp.stanford.edu/linguistics/sag/bms-nllt.ps.

Bresnan, J. (ed.) (1982). *The mental representation of grammatical relations*. Cambridge, MA: MIT Press.

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

Chomsky, N. (1973). Conditions on Transformations. In S. A. Anderson and P. Kiparsky (eds.), *A Festschrift for Morris Halle*, New York: Holt, Rinehart and Winston.

Chomsky, N. (1977). On *wh*–movement. In P. Culicover, T. Wasow and A. Akmajian (eds.), *Formal Syntax*, Academic Press, pp. 71–132.

Chomsky, N. (1981). *Lectures on government and binding*. Dordercht: Foris.

Chomsky, N. (1982). *Concepts and consequences of the theory of government and binding*. Cambridge: MIT Press.

Chomsky, N. (1986). *Barriers*. Cambridge, MA: MIT Press.

Chomsky, N. (1995). *The minimalist program*. Cambridge, MA: MIT Press.

Cinque, G. (1990). *Types of $\bar{A}$–dependencies*. Cambridge, MA: MIT Press.

Contreras, H. (1984). A note on parasitic gaps. *Linguistic Inquiry* 15, 698–701.

Culicover, P. (1976). *Syntax*. New York: Academic Press.

De Kuthy, K. and Meurers, W. D. (2001). On partial constituent fronting in German. *Journal of Comparative Germanic Linguistics* 3(3), 143–205. http://ling.osu.edu/~dm/papers/dekuthy-meurers-jcgl01.html.

Fanselow, G. (1987). *Konfigurationalität*. Tübingen: Gunter Narr.

Fanselow, G. (1988). Aufspaltung von NPn und das Problem der 'freien' Wortstellung. *Linguistische Berichte* 114, 91–113.

Frampton, J. (1990). Parasitic gaps and the theory of *wh*-chains. *Linguistic Inquiry* 21, 49–77.

Gazdar, G. (1981). Unbounded dependencies and coordinate structure. *Linguistic Inquiry* 12, 155–84.

Gazdar, G., Klein, E., Pullum, G. K. and Sag, I. (1985). *Generalized Phrase Structure Grammar*. Cambridge, MA: Harvard University Press.

Gazdar, G., Pullum, G. K., Sag, I. and Wasow, T. (1982). Coordination and transformational grammar. *Linguistic Inquiry* 13, 663–676.

Goodall, G. (1987). *Parallel Structures in Syntax*. Cambridge, UK: Cambridge University Press.

Hukari, T. E. and Levine, R. D. (1995). Adjunct extraction. *Journal of Linguistics* 31, 195–226.

Jacobson, P. and Johnson, M. (1991). Tree generation with context-sensitive rules yields only context-free languages. In A. Halpern (ed.), *The proceedings of the Ninth West Coast Conference on Formal Languages (WCCFL 9)*, Stanford, CA: CSLI, pp. 289–303.

Johnson, D. and Lappin, S. (1999). *Local constraints vs. economy*. Stanford, CA: CSLI Publications.

Johnson, M. (1994). Two ways of formalizing grammars. *Linguistics and Philosophy* 17, 221–248.

Kayne, R. (1994). *Connectedness and Binary Branching*. Dordrecht: Foris.

Kearney, K. (1983). Governing categories. Unpublished ms. University of Connecticut, Storrs, CT.

Kuhn, J. (2001). Resource sensitivity in the syntax-semantics interface: Evidence from the German Split NP construction. In W. D. Meurers and T. Kiss (eds.), *Constraint-based approaches to Germanic syntax*, Stanford, CA: CSLI Publications, Studies in Constraint-Based Lexicalism, pp. 177–215.

Levine, R., Hukari, T. E. and Calcagno, M. (2000). Some overlooked parasitic gap constructions in English and their theoretical implications. In P. Culicover and P. M. Postal (eds.), *Parasitic Gaps*, Cambridge, MA: MIT Press, pp. 181–222.

Levine, R. D. (2001). The extraction riddle: Just what are we missing? *Journal of Linguistics* 37, 145–174.

Levine, R. D. (2004). The syntax of extraction: Derivation or constraint satisfaction? In O. Bonami and P. C. Hofherr (eds.), *Empirical Issues in Formal Syntax and Semantics 5*, Paris: CSSP, vol. 5, pp. 159–178. http://www.cssp.cnrs.fr/eiss5/index_en.html.

Levine, R. D. and Hukari, T. E. (to appear). *The unity of unbounded dependency constructions*. Stanford, CA: CSLI Publications.

Levine, R. D. and Sag, I. A. (2003). *Wh*-nonmovement. *Gengo Kenkyu. Journal of the Linguistic Society of Japan* 123, 171–220.

Matthews, G. H. (1961). Analysis by synthesis of sentences of natural language. In *First International Conference on Machine Translation*, Teddington, England.

McCawley, J. (1968). Concerning the base component of a transformational grammar. *Foundations of Language* 4(3), 243–269.

Müller, G. (1991). Abstrakte inkorporation. In S. Olsen and G. Fanselow (eds.), *DET, COMP and INFL*, Tübingen: Max Niemeyer, pp. 155–202.

Müller, G. (1996). Incomplete category fronting. Habilitationsschrift, Universität Tübingen. Available as SfS-Report 01-96.

Munn, A. (2001). Explaining parasitic gap restrictions. In P. Culicover and P. Postal (eds.), *Parasitic Gaps*, Cambridge, MA: MIT Press.

Oehrle, D. (2000). Context-sensitive node admissibility. *Grammars* 3(2–3), 275–293.

Perrault, C. R. (1984). On the mathematical properties of linguistic theories. *Computational Linguistics* 10(3–4), 165–176.

Peters, S. and Ritchie, R. W. (1973). Context-sensitive immediate constituent analysis: Context-free languages revisited. *Mathematical Systems Theory* 6(4), 324–333.

Pollard, C. and Sag, I. A. (1987). *Information-based syntax and semantics, vol. 1: Fundamentals*. No. 13 in CSLI Lecture Notes. Stanford, CA: CSLI Publications.

Pollard, C. and Sag, I. A. (1994). *Head-Driven Phrase Structure Grammar*. Chicago, IL: University of Chicago Press.

Postal, P. M. (1993b). Remarks on weak crossover. *Linguistic Inquiry* 24, 539–556.

Pullum, G. K. and Scholz, B. C. (2001). On the distinction between model-theoretic and generative-enumerative syntactic frameworks. In P. de Groote, G. Morrill and C. Retoré (eds.), *Logical aspects of computational linguistics:*

4th international conference, Berlin: Springer Verlag, no. 2099 in Lecture Notes in Artificial Intelligence, pp. 17–43.

Thiersch, C. (1985). VP and scrambling in the German Mittelfeld. Ms., Universität Köln and University of Conneticut, Storrs, CT.

van Geenhoven, V. (1998). *Semantic incorporation and indefinite descriptions: Semantic and syntactic aspects of noun incorporation in West Greenlandic*. Stanford, CA: CSLI Publications.

van Riemsdijk, H. (1989). Movement and regeneration. In P. Benicà (ed.), *Dialect variation and the theory of grammar*, Dordrecht: Foris.

Webelhuth, G. and den Besten, H. (1987). Remnant topicalization and the constituent structure of VP in the Germanic SOV languages (abstract for the 10th GLOW Colloquium). In H. Bennis and J. Koster (eds.), *GLOW Newsletter Nr. 18*, Dordrecht: Foris, pp. 15–16.

Zaenen, A. (1983). On syntactic binding. *Linguistic Inquiry* 14, 469–504.