Research Report AI-1992-03

GB Theory as Dependency Grammar

Michael A. Covington
Artificial Intelligence Programs
The University of Georgia
Athens, Georgia 30602 U.S.A.

GB THEORY AS DEPENDENCY GRAMMAR

Michael A. Covington

Artificial Intelligence Programs
The University of Georgia
Athens, Georgia 30602, USA
email: mcovingt@uga.cc.uga.edu

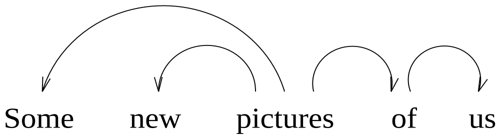
Presented at the International Congress of Linguists,
Québec, August 1992

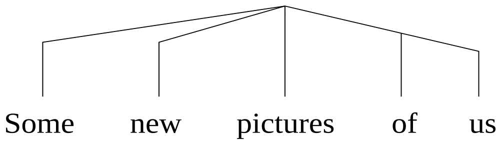
There are two major ways to describe syntactic structure: break the sentence into substrings (**constituents**), or establish links between individual words (**dependencies**). Government-binding (GB) theory is largely a dependency theory even though it still uses constituency notation. GB theory can be simplified by recasting it into dependency formalism.

1 Constituency vs. dependency

Both constituency grammar and dependency grammar are centuries old. The concept of dependency apparently originated with the Arabs and was adopted into Latin traditional grammar in the Middle Ages (Covington 1984). Constituent analysis can be traced back to the analysis of logical propositions by the Stoics (Mates 1961).

The key claim of dependency grammar (DG) is that all constructions are **endocentric**, i.e., every phrase has a most prominent element (the **head**) which determines its syntactic properties. Modifiers and complements of the head are called **dependents**. Structure can be diagrammed by drawing arrows from heads to dependents, or by drawing a "dependency tree" (D-tree) in which heads are attached to dependents by lines sloping downward:





Here we say that *some, new,* and *of* are **immediate** or **direct** dependents, and *us* is an **indirect** dependent, of *pictures.*

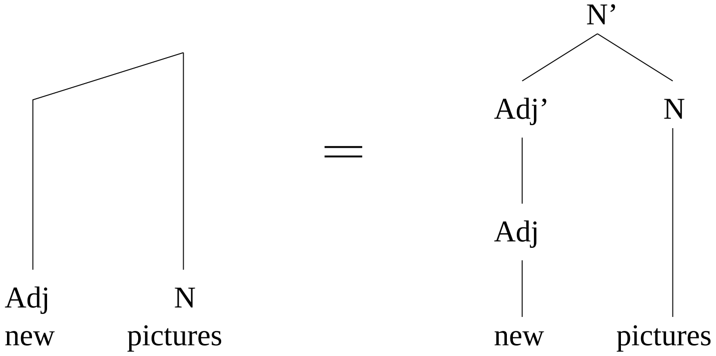
In dependency grammar, constituents still exist, but they are a derived rather than a basic concept. A constituent consists of any head plus all its direct and indirect dependents.

As Mel′čuk (1987) has emphasized, dependency grammar is not a **theory** of language but rather a **notation** for describing structure. Theories of grammar can of course be built upon it, as Mel′čuk, Tesnière, Hudson, Starosta, and others have done.

2 Dependency and X-bar theory

2.1 Dependency trees correspond to X-bar trees

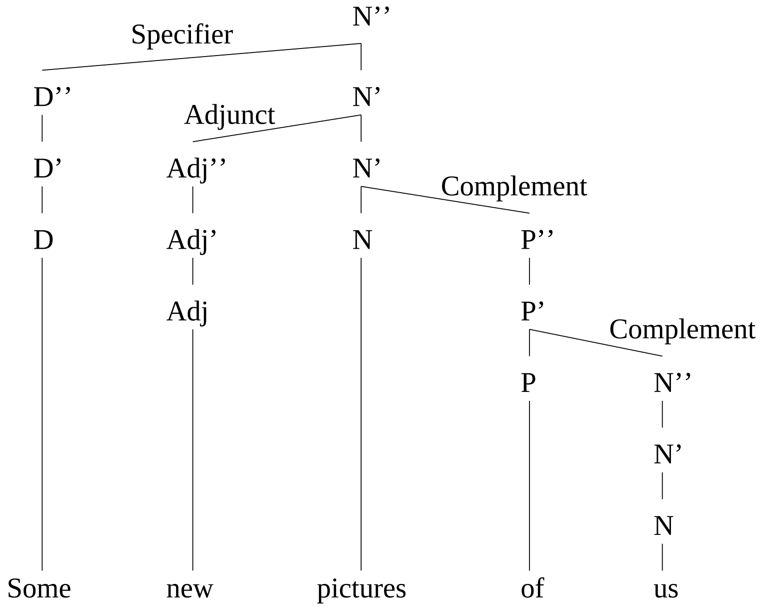
GB theory shares with DG the crucial claim that every phrase has a head. Dependency trees are equivalent to X-bar phrase-structure trees with only one non-terminal bar level, like this:



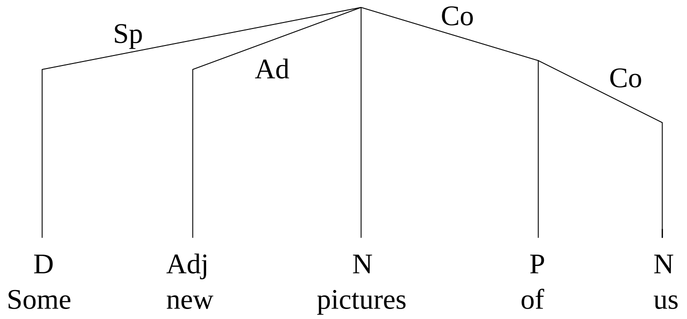
Since the D-tree contains no non-branching nodes, the Adj′ node here has no direct counterpart in it; rather, Adj′ is supplied implicitly by X-bar theory, which stipulates that all sisters of the head are maximal projections.

2.2 GB theory recognizes three kinds of dependency

One non-terminal bar level is not enough for current GB theory. GB theory recognizes 3 kinds of sisters of the head: **complements** (sisters of X), **adjuncts** (sisters of X′ dominated by X′), and the **specifier** (the sister of X′ dominated by X″), as in this example:



D-trees cannot preserve the distinction between X, X′, and X″ which GB theory uses to distinguish between complements, adjuncts, and specifiers. Instead, I propose that complement, adjunct, and specifier be treated as three kinds of dependency, and labeled as such in the D-tree:



Word order rules for English can then specify that:

(1) Every constituent (= head plus all direct and indirect dependents) is continuous in linear order.

(2) Complements come after the head, before all other dependents.

(3) Adjuncts which are adjective phrases precede the head;

(4) Adjuncts which are prepositional phrases follow the head;

(5) Specifiers precede the head and all its other dependents;

and so on.

3 Government as dependency

Adopting dependency formalism simplifies the definition of (head) government, which, in GB theory, is the relation between heads and the words to which they assign case, thematic roles, and/or agreement features. Chomsky (1986) defines government essentially as follows:

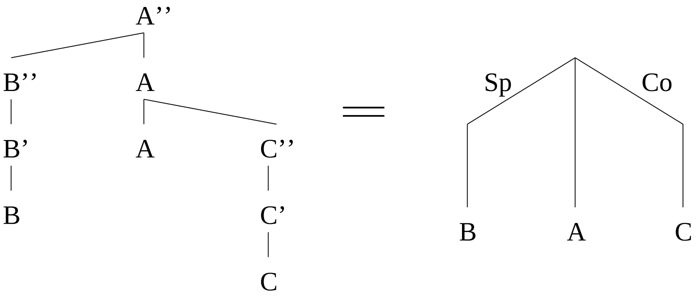
A governs B if and only if:

(1) A does not dominate B;

(2) the lowest maximal projection that dominates A also dominates B;

(3) there is no maximal projection (of a lexical head) between A and B.[1]

Consider now the structure:



If only lexical items can govern, then A governs B$''$ and C$''$. If phrases can also govern, then B$''$ governs A$'$, A, and C$''$, and C$''$ governs A$'$, A, and B$''$.

With the tree recast into dependency notation, it is clear that if only lexical items can govern, then the definition of government is:

A governs B iff B is an immediate dependent of A.

One can hardly ask for this to be simpler. If phrases can also govern then:

A governs B iff B is an immediate dependent of A or of the head of A.

Recent work (Rizzi 1990) suggests that government by phrases is in several ways different from government by lexical items. If this is so, the clumsiness in the second definition of government arises from the fact that it conflates two different grammatical relations, both of which pattern according to dependency, not constituent structure.

[1] Assuming IP and CP are not inherent barriers because they do not have lexical heads.

References

[1] Chomsky, N. (1986) *Barriers.* Cambridge, Mass.: MIT Press.

[2] Covington, M. A. (1984) *Syntactic Theory in the High Middle Ages.* Cambridge, England: Cambridge University Press.

[3] Hudson, R. A. (1991) *English Word Grammar.* Oxford: Blackwell.

[4] Hudson, R. A. (1988) Coordination and grammatical relations. *Journal of Linguistics*, 24, 303–342.

[5] Mates, B. (1961) *Stoic Logic*. Berkeley: U. of California Press.

[6] Mel′čuk, I. A. (1987) *Dependency Syntax: Theory and Practice.* Buffalo: SUNY Press.

[7] Rizzi, L. (1990) *Relativized Minimality.* Cambridge, Mass.: MIT Press.

[8] Starosta, S. (1988) *The Case for Lexicase.* London: Pinter.

[9] Tesnière, L. (1959) *Élements de syntaxe structurale*. Paris: Klincksieck.